future internet

Special Issue Reprint

Cybersecurity in the IoT

Edited by
Christos Tryfonopoulos and Nicholas Kolokotronis

mdpi.com/journal/futureinternet

MDPI

Cybersecurity in the IoT

Cybersecurity in the IoT

Guest Editors

Christos Tryfonopoulos
Nicholas Kolokotronis

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Guest Editors

Christos Tryfonopoulos
Department of Informatics
and Telecommunications
University of the Peloponnese
Tripoli
Greece

Nicholas Kolokotronis
Department of Informatics
and Telecommunications
University of the Peloponnese
Tripoli
Greece

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Special Issue, published open access by the journal *Future Internet* (ISSN 1999-5903), freely accessible at: https://www.mdpi.com/journal/futureinternet/special_issues/7W4G8W8I4T.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-9063-7 (Hbk)
ISBN 978-3-7258-9064-4 (PDF)
https://doi.org/10.3390/books978-3-7258-9064-4

© 2026 by the authors. Articles in this reprint are Open Access and distributed under the Creative Commons Attribution (CC BY) license. The reprint as a whole is distributed by MDPI under the terms and conditions of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) license (https://creativecommons.org/licenses/by-nc-nd/4.0/).

Contents

About the Editors . . . . . **vii**

Christos Tryfonopoulos and Nicholas Kolokotronis
Cybersecurity in the IoT
Reprinted from: *Future Internet* **2026**, *17*, 127, https://doi.org/10.3390/fi18030127 . . . . . **1**

Elias Dritsas and Maria Trigka
A Survey on Cybersecurity in IoT
Reprinted from: *Future Internet* **2025**, *17*, 30, https://doi.org/10.3390/fi17010030 . . . . . **4**

Emiliia Geloczi, Felix Klement, Patrick Struck and Stefan Katzenbeisser
SoK: Delegated Security in the Internet of Things
Reprinted from: *Future Internet* **2025**, *17*, 202, https://doi.org/10.3390/fi17050202 . . . . . **37**

Samaneh Madanian, Tserendorj Chinbat, Maduka Subasinghage, David Airehrour, Farkhondeh Hassandoust and Sira Yongchareon
Health IoT Threats: Survey of Risks and Vulnerabilities
Reprinted from: *Future Internet* **2024**, *16*, 389, https://doi.org/10.3390/fi16110389 . . . . . **91**

Amr Adel and Tony Jan
Watch the Skies: A Study on Drone Attack Vectors, Forensic Approaches, and Persisting Security Challenges
Reprinted from: *Future Internet* **2024**, *16*, 250, https://doi.org/10.3390/fi16070250 . . . . . **124**

Mohammed N. Alenezi
Significance of Machine Learning-Driven Algorithms for Effective Discrimination of DDoS Traffic Within IoT Systems
Reprinted from: *Future Internet* **2025**, *17*, 266, https://doi.org/10.3390/fi17060266 . . . . . **146**

Bo Cui, Yachao Chai, Zhen Yang and Keqin Li
Intrusion Detection in IoT Using Deep Residual Networks with Attention Mechanisms
Reprinted from: *Future Internet* **2024**, *16*, 255, https://doi.org/10.3390/fi16070255 . . . . . **172**

Prabu Kaliyaperumal, Sudhakar Periyasamy, Manikandan Thirumalaisamy, Balamurugan Balusamy and Francesco Benedetto
A Novel Hybrid Unsupervised Learning Approach for Enhanced Cybersecurity in the IoT
Reprinted from: *Future Internet* **2024**, *16*, 253, https://doi.org/10.3390/fi16070253 . . . . . **187**

Fouzi Harrou, Benamar Bouyeddou, Abdelkader Dairi and Ying Sun
Exploiting Autoencoder-Based Anomaly Detection to Enhance Cybersecurity in Power Grids
Reprinted from: *Future Internet* **2024**, *16*, 184, https://doi.org/10.3390/fi16060184 . . . . . **216**

Antonio Maña, Francisco J. Jaime and Lucía Gutérrez
A Secure Auditable Remote Registry Pattern for IoT Systems
Reprinted from: *Future Internet* **2024**, *16*, 405, https://doi.org/10.3390/fi16110405 . . . . . **236**

Tuan-Kiet Dang, Khai-Duy Nguyen, Binh Kieu-Do-Nguyen, Trong-Thuc Hoang and Cong-Kha Pham
Realization of Authenticated One-Pass Key Establishment on RISC-V Micro-Controller for IoT Applications
Reprinted from: *Future Internet* **2024**, *16*, 157, https://doi.org/10.3390/fi16050157 . . . . . **254**

Aryan Sharma, Haoming Wang, Deepak Mishra and Aruna Seneviratne
Conservative Interference Injection to Minimize Wi-Fi Sensing Privacy Risks and Bandwidth Loss
Reprinted from: *Future Internet* **2025**, *17*, 20, https://doi.org/10.3390/fi17010020 . . . . . **271**

About the Editors

Christos Tryfonopoulos

Christos Tryfonopoulos currently serves as a Professor and Vice-Head of the Dept. of Informatics & Telecommunications, University of the Peloponnese. He holds a PhD and an MSc in Computer Engineering from the Dept. of Electrical and Computer Engineering, Technical University of Crete and a BSc in Computer Science from the Dept. of Computer Science, University of Crete. In the past, Christos has served as Director of the joint MSc program in Space Science Technologies and Applications, between the University of the Peloponnese and the National Observatory of Athens; has coordinated the P2P and Information Management group at the Databases and Information Systems Department, Max Planck Institute for Informatics in Saarbruecken, Germany; and has worked as a research associate at the Intelligent Systems Lab, Dept. of Electrical and Computer Engineering, Technical University of Crete. Christos' research interests include machine learning applications, data/information management and large-scale distributed systems. He has published more than 100 papers in book chapters, international journals, conferences and workshops, and served as an editor/PC member in top-ranked journals/conferences. He has extensive experience and active involvement in R&D projects; he has led and/or participated in 20+ competitive EU (HORIZON 2020, IST/FET) and national (GSRT, NSRF) projects; and he is actively serving as an external evaluator for EU and national bodies. He has received two international scholarships and two best paper awards at international conferences, participated in international promotion committees, and has given invited talks on a wide range of topics in top-tier institutions including TUWien, CWI, L3S, MPII, KAUST and others.

Nicholas Kolokotronis

Nicholas Kolokotronis is a full Professor in Cryptography and Security and the Director of the Cryptography and Security Group at the Department of Informatics and Telecommunications, University of the Peloponnese. He received a B.Sc. in Mathematics in 1995 from Aristotle University of Thessaloniki, Greece, as well as an M.Sc. (highest honors) on highly efficient algorithms in 1998 and a Ph.D. (highest honors) in Cryptography in 2003, both from the National and Kapodistrian University of Athens. He has been a faculty member at the University of the Peloponnese since 2008, and he has served as a visiting professor at the University of Piraeus, the University of the Peloponnese, the National and Kapodistrian University of Athens, and the Open University of Cyprus since 2004. He has supervised more than 75 Ph.D., M.Sc., and B.Sc. students, who were guided into research activities. He has participated in more than 25 EU-funded research and innovation projects and national projects, the total funding of which is over 55 M€. He has more than 130 publications in international scientific journals, conferences, and books. He is an editor and the lead author of the CRC Press books *Internet of Things, Threats, Landscape, and Countermeasures* and *Cyber-Security Threats, Actors, and Dynamic Mitigation*, both released in 2021. He has been a Guest Editor in various Springer and MDPI journals, an Associate Editor of the EURASIP *Journal on Wireless Communications and Networking* (2009–17) and a regular reviewer for prestigious journals, incl. *IEEE IT*, *IEEE TIFS*, *IEEE TETC*, *IEEE TIT*, Springer's *DCC*, Elsevier's *COSE*, etc. He has served as a chair of international conferences (IEEE CSR 2021–26) and workshops (IEEE SecSoft 2019/21, IEEE CSRIoT 2019–20, and ACM EPESec 2020–23) focusing on various facets of security, and a TPC member in many international conferences, incl. IEEE ISIT, IEEE GLOBECOM, IEEE ICC, ISC, etc. His research interests span the broad areas of security and cryptography.

future internet

MDPI

Editorial

Cybersecurity in the IoT

Christos Tryfonopoulos * and Nicholas Kolokotronis

Department of Informatics and Telecommunications, University of the Peloponnese, GR 22100 Tripoli, Greece; nkolok@uop.gr

* Correspondence: trifon@uop.gr

The Internet of Things (IoT) has evolved into a vast ecosystem of massively interconnected devices delivering intelligent services across consumer, commercial, and industrial environments. Despite its transformative potential, the IoT remains inherently vulnerable. Many deployed devices are resource-constrained and frequently designed without robust security primitives, often lacking strong authentication, secure update mechanisms, or proper credential management. The integration of IoT infrastructures with cloud platforms and distributed services further increases architectural complexity, reduces visibility, and expands the attack surface. Modern botnets increasingly weaponize both IoT devices and cloud resources, while supply chain compromises and zero-trust violations expose systemic weaknesses in traditionally perimeter-based security models. In this context, intelligent, adaptive, and cross-layer security approaches are no longer optional but essential for effective preparation, prevention, detection, and mitigation of cyber-threats.

This Special Issue aims at presenting state-of-the-art methodologies, algorithms, analyses, protocols, and frameworks that address cybersecurity challenges in IoT ecosystems. The accepted contributions reflect the breadth of the original call, covering AI-driven security techniques, architectural and protocol-level innovations, domain-specific cybersecurity analyses, and disruptive defensive mechanisms spanning device, network, and cloud-integrated environments. To present a coherent perspective on the field's evolution, the accepted papers are organized into three thematic groups. The first group establishes the threat landscape and formal foundations of IoT security through surveys, systematization efforts, and domain-specific analyses. The second group focuses on AI-based intrusion and anomaly detection mechanisms that enable intelligent and adaptive defense in large-scale IoT and cyber–physical systems, while the third group of papers highlights secure-by-design architectures, lightweight cryptographic protocols, and privacy-preserving mechanisms that embed security directly into IoT infrastructures. This grouping reflects a logical research trajectory—from understanding vulnerabilities to detecting adversarial behavior and engineering resilient and trustworthy IoT systems—thereby capturing the multidimensional nature of cybersecurity in the Future Internet.

The first group of contributions establishes the conceptual and analytical foundations of IoT cybersecurity by systematizing knowledge, formalizing security notions, and examining domain-specific threat landscapes. The survey [1] provides a general yet comprehensive overview of vulnerabilities, architectural weaknesses, and mitigation strategies across the IoT ecosystem, offering a holistic synthesis of device, network, and data-layer security challenges. Complementing this broad perspective, the work in [2] delivers a knowledge systematization on delegated security in IoT, introducing formalized security notions and a mathematical interpretation of the confidentiality/integrity/availability triad within constrained environments. By analyzing delegation as a design paradigm, this work strengthens the theoretical underpinnings of resource-aware IoT protection. At the

domain level, the authors in [3] focus on healthcare IoT, presenting a layered taxonomy of threats across perception, network, and application layers while emphasizing regulatory and privacy implications in safety-critical environments. Finally, the work in [4] extends the discussion to Unmanned Aerial Vehicle (UAV) ecosystems, examining drone-specific threat models, forensic artifacts, and persistent operational challenges. Together, these works map the evolving attack surface of IoT systems, highlight the need for domain-aware threat modeling and formal reasoning, and define the risk landscape that motivates the technical innovations presented in the rest of the contributions of this Special Issue.

The second group reflects the growing centrality of artificial intelligence in defending IoT infrastructures. These contributions collectively explore supervised, unsupervised, and hybrid learning paradigms for intrusion detection and anomaly identification in dynamic and heterogeneous environments. The work in [5] evaluates multiple machine learning and deep learning classifiers for DDoS discrimination, emphasizing the importance of feature selection and preprocessing for performance optimization, while [6] advances deep intrusion detection architectures by integrating residual temporal convolutional modules with attention mechanisms; this allows for improved feature extraction and generalization capability across benchmark datasets. Moving toward reduced reliance on labeled data, refs. [7,8] adopt semi-supervised and unsupervised paradigms centered on autoencoder-based feature extraction. The work in [7] proposes a hybrid framework combining basic and deep autoencoders with one-class classification and density-based clustering, enabling detection of previously unseen attacks while addressing data imbalance. Similarly, the authors in [8] apply a GRU-enhanced stacked autoencoder in smart grid environments, coupling deep representation learning with anomaly detection algorithms to secure cyber–physical energy infrastructures. Collectively, this group of contributions illustrates a methodological shift toward adaptive, data-driven security intelligence capable of addressing the scale, heterogeneity, and evolving attack patterns of modern IoT deployments via AI-driven approaches.

While detection is critical, proactive security engineering remains relevant; the third group of works presents constructive approaches that embed security and privacy directly into IoT architectures, protocols, and communication layers. The work in [9] introduces a secure and auditable registry architectural pattern based on message-oriented middleware, emphasizing reusability, real-world applicability, and integration into broader security pattern languages. This contribution highlights the importance of auditability and accountability in large-scale IoT systems transitioning from legacy infrastructures. At the protocol level, the authors of [10] propose a lightweight authenticated one-pass key establishment mechanism implemented on a micro-controller with Reduced Instruction Set Computing (RISC-V) architecture. By combining elliptic curve cryptographic schemes and optimizing for constrained devices, the work demonstrates how strong security guarantees can be achieved with minimal computational overhead. Complementing these architectural and cryptographic contributions, the work in [11] addresses privacy risks emerging from next-generation Wi-Fi sensing capabilities. By modeling channel contention and introducing conservative interference injection as a defense strategy, it provides a physical-layer privacy-preserving mechanism that balances sensing degradation with bandwidth efficiency. Altogether, this group of papers emphasizes that IoT cybersecurity must be engineered across layers—from hardware and cryptography to middleware and wireless communication.

Taken together, the contributions in this Special Issue outline a coherent research trajectory for cybersecurity in the IoT era. Initially, a rigorous understanding of the threat landscape and formal security foundations to characterize vulnerabilities across diverse and domain-specific IoT ecosystems is provided via the works in [1–4]. As IoT environments

https://doi.org/10.3390/fi18030127

grow in scale and complexity, adaptive and intelligent detection mechanisms [5–8] that leverage machine/deep learning focus on identifying evolving attack patterns in real time. Finally, secure-by-design architectures, lightweight cryptographic protocols, and privacy-preserving mechanisms shift the focus from reactive detection to embedding IoT cybersecurity at the system level via hardware validation and physical-layer privacy defenses. This progression, from understanding risk to detecting adversarial behavior and engineering resilient systems, captures the multidimensional nature of IoT security and highlights the maturing of the research field: moving beyond isolated solutions towards integrated, cross-layer, and intelligence-driven approaches capable of securing the Future Internet.

Author Contributions: All authors contributed equally to this editorial. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Dritsas, E.; Trigka, M. A Survey on Cybersecurity in IoT. *Future Internet* **2025**, *17*, 30. [CrossRef]
2. Geloczi, E.; Klement, F.; Struck, P.; Katzenbeisser, S. SoK: Delegated Security in the Internet of Things. *Future Internet* **2025**, *17*, 202. [CrossRef]
3. Madanian, S.; Chinbat, T.; Subasinghage, M.; Airehrour, D.; Hassandoust, F.; Yongchareon, S. Health IoT Threats: Survey of Risks and Vulnerabilities. *Future Internet* **2024**, *16*, 389. [CrossRef]
4. Adel, A.; Jan, T. Watch the Skies: A Study on Drone Attack Vectors, Forensic Approaches, and Persisting Security Challenges. *Future Internet* **2024**, *16*, 250. [CrossRef]
5. Alenezi, M.N. Significance of Machine Learning-Driven Algorithms for Effective Discrimination of DDoS Traffic Within IoT Systems. *Future Internet* **2025**, *17*, 266. [CrossRef]
6. Cui, B.; Chai, Y.; Yang, Z.; Li, K. Intrusion Detection in IoT Using Deep Residual Networks with Attention Mechanisms. *Future Internet* **2024**, *16*, 255. [CrossRef]
7. Kaliyaperumal, P.; Periyasamy, S.; Thirumalaisamy, M.; Balusamy, B.; Benedetto, F. A Novel Hybrid Unsupervised Learning Approach for Enhanced Cybersecurity in the IoT. *Future Internet* **2024**, *16*, 253. [CrossRef]
8. Harrou, F.; Bouyeddou, B.; Dairi, A.; Sun, Y. Exploiting Autoencoder-Based Anomaly Detection to Enhance Cybersecurity in Power Grids. *Future Internet* **2024**, *16*, 184. [CrossRef]
9. Maña, A.; Jaime, F.J.; Gutiérrez, L. A Secure Auditable Remote Registry Pattern for IoT Systems. *Future Internet* **2024**, *16*, 405. [CrossRef]
10. Dang, T.K.; Nguyen, K.D.; Kieu-Do-Nguyen, B.; Hoang, T.T.; Pham, C.K. Realization of Authenticated One-Pass Key Establishment on RISC-V Micro-Controller for IoT Applications. *Future Internet* **2024**, *16*, 157. [CrossRef]
11. Sharma, A.; Wang, H.; Mishra, D.; Seneviratne, A. Conservative Interference Injection to Minimize Wi-Fi Sensing Privacy Risks and Bandwidth Loss. *Future Internet* **2025**, *17*, 20. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Review

A Survey on Cybersecurity in IoT

Elias Dritsas * and Maria Trigka

Industrial Systems Institute (ISI), Athena Research and Innovation Center, 26504 Patras, Greece; trigka@isi.gr
* Correspondence: dritsas@isi.gr

Abstract: The proliferation of the Internet of Things (IoT) has transformed the digital landscape, enabling a vast array of interconnected devices to communicate and share data seamlessly. However, the rapid expansion of IoT networks has also introduced significant cybersecurity challenges. This paper presents a comprehensive survey of cybersecurity in the IoT ecosystem, examining the current state of research, identifying critical security vulnerabilities, and exploring advanced strategies for mitigating threats. The survey covers various facets of IoT security, including device authentication, data integrity, privacy, network security, and the emerging role of artificial intelligence (AI) in bolstering cybersecurity defenses. By synthesizing existing research and highlighting ongoing challenges, this survey aims to provide a holistic understanding of IoT cybersecurity and to guide future research endeavors.

Keywords: Internet of Things; cybersecurity; network security; data privacy; blockchain security

1. Introduction

The IoT has emerged as a transformative force in modern technology, driving innovation across various sectors by enabling seamless connectivity among a vast array of devices, sensors, and systems. From industrial automation to smart healthcare, IoT is reshaping industries by facilitating real-time data collection, analysis, and decision-making, leading to enhanced operational efficiency, reduced costs, and the creation of new business models. However, the very characteristics that make IoT so powerful—its distributed nature, the heterogeneity of devices, and pervasive connectivity—also introduce unprecedented cybersecurity challenges [1–3].

Unlike traditional information technology (IT) systems, IoT networks consist of diverse devices, ranging from powerful servers to tiny, resource-constrained sensors, all of which must operate harmoniously in often untrusted environments. This heterogeneity presents unique security challenges, as each device type may require different protection mechanisms. Moreover, many IoT devices are designed with minimal security features due to cost constraints or limited computational resources, making them susceptible to various cyber threats. The consequences of such vulnerabilities can be severe, ranging from the compromise of individual devices to large-scale disruptions in critical infrastructure systems, potentially leading to catastrophic outcomes in sectors such as healthcare, energy, and transportation [4–6].

Furthermore, the IoT ecosystem is characterized by its extensive use of wireless communication, which, while enabling flexibility and scalability, also exposes networks to a wider range of attack vectors. Wireless communication channels are inherently less secure

than wired ones, making them more vulnerable to eavesdropping, man-in-the-middle attacks, and jamming. Additionally, the dynamic nature of IoT environments, where devices frequently join and leave the network, further complicates the implementation of traditional security protocols, which are often designed for static, well-defined network architectures [7–9].

Another critical aspect of IoT security is the data it generates and processes. IoT devices continuously collect vast amounts of data, much of which are sensitive or private. The protection of these data throughout its lifecycle—from collection and transmission to storage and processing—is paramount. However, ensuring data confidentiality, integrity, and availability in IoT systems is challenging due to the decentralized nature of data storage and the reliance on third-party cloud services for data processing. Moreover, IoT systems often involve multiple stakeholders, each with different security requirements and risk tolerances, complicating the development of unified security policies [10–12].

The rapid evolution of IoT technology has outpaced the development of comprehensive security frameworks, resulting in a landscape where security considerations are often an afterthought rather than a foundational design principle. This oversight has led to the proliferation of insecure devices and networks, which are now being exploited by attackers for various malicious purposes, including data theft, unauthorized surveillance, and the orchestration of large-scale distributed denial-of-service (DDoS) attacks [13–15].

In response to these challenges, the field of IoT cybersecurity has seen a surge in research efforts aimed at developing innovative solutions that address the unique security needs of IoT environments. These efforts include the design of lightweight cryptographic algorithms tailored for resource-constrained devices, the development of decentralized security models such as blockchain, and the application of AI and machine learning (ML) to enhance threat detection and response capabilities [16,17].

The present survey distinguishes itself by addressing the cybersecurity challenges in IoT through a more comprehensive and integrative approach compared to the surveys summarized in Table 1. Unlike prior works that focus narrowly on specific aspects such as blockchain, ML applications, or industry-specific IoT environments, this survey provides a broader scope by synthesizing diverse cybersecurity dimensions, including device authentication, data integrity, privacy, network security, and the role of AI in enhancing security. Furthermore, it delves into emerging technologies and methodologies, such as federated learning, post-quantum cryptography, and decentralized identity management systems, which are only briefly touched upon or overlooked in existing studies. By offering an extensive analysis of both technical and operational challenges, this work serves as an essential resource for developing scalable, adaptive, and future-proof IoT security solutions. In conclusion, this submission contributes to the field of IoT cybersecurity in several significant ways:

- It provides a comprehensive overview of the unique cybersecurity challenges specific to IoT environments.
- It examines advanced strategies for IoT device authentication, data integrity, and privacy protection.
- It analyzes the security implications of various IoT communication protocols and network architectures.
- It explores the application of AI and ML in enhancing IoT cybersecurity measures.
- It identifies key areas for future research to address ongoing and emerging IoT security challenges.

The remaining paper is illustrated in Figure 1 and structured as follows. Section 2 notes the importance of device authentication and identity management. Next, Section 3 analyzes the challenges of data integrity and privacy in the IoT. Section 4 refers to network security and communication protocols. Moreover, Section 5 outlines the integration of AI and ML into IoT security. Section 6 discusses challenges and future directions. Finally, Section 7 summarizes the findings of this study.

Table 1. Descriptive summary of related surveys on cybersecurity in IoT.

Survey	Description
[18]	Overview of techniques and elements to achieve cybersecurity in blockchain-based systems. Includes lessons learned from analyzing academic papers and industrial applications. Highlights gaps and research opportunities.
[19]	Examines IoT cybersecurity frameworks, emphasizing communication protocols, cybersecurity challenges, and intrusion detection systems (IDSs). Presents a validated IoT cybersecurity framework and discusses its role in protecting data and systems.
[20]	Investigates the application of ML in static analysis for IoT cybersecurity. Focuses on automating and intellectualizing the analysis of heterogeneous IoT systems and proposes an intelligent framework.
[21]	Explores learning-based methods for detecting cyberattacks in IoT systems. Discusses ML and deep learning (DL) approaches for various attacks, including DDoS, spoofing, and man-in-the-middle attacks.
[22]	Surveys classifications and mitigations of cyberattacks on IoT and industrial IoT devices, focusing on Industry 4.0. Highlights the integration of IoT with supervisory control and data acquisition systems in manufacturing industries and related vulnerabilities.
[23]	Discusses cybersecurity threats in IoT-enabled maritime industries, addressing risks in maritime operations, confidentiality, and integrity. Analyzes risk mitigation strategies and frameworks for safeguarding critical systems.
[24]	Notes cybersecurity certification schemes for IoT, analyzing challenges in developing frameworks that integrate risk assessment, security requirements, and governance. Provides insights into certification for emerging IoT scenarios.
[25]	Comprehensive survey of cybersecurity in IoT-based cloud computing, focusing on threats, cloud architecture, and the integration of AI and DL to enhance security. Covers challenges related to data, network, and application layers.

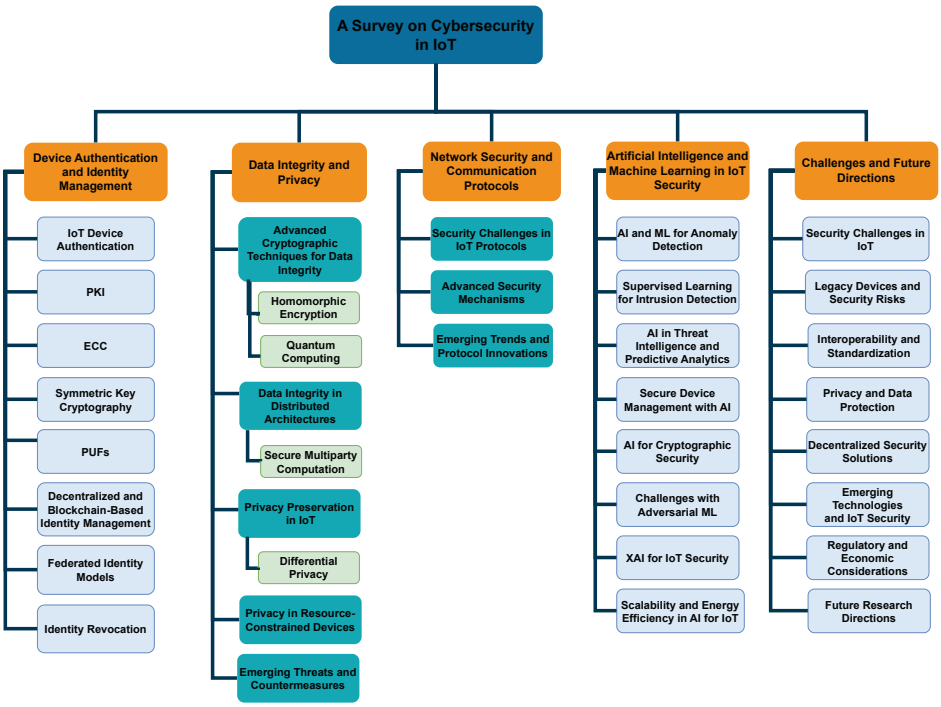


Figure 1. Illustrative diagram of the survey's structure.

2. Device Authentication and Identity Management

Device authentication and identity management are critical components of the security architecture in the IoT ecosystem, serving as the first line of defense against unauthorized access and malicious activities [26]. Unlike traditional computing environments, IoT networks are characterized by a massive scale of heterogeneously connected devices, each with varying levels of computational power and security capabilities [27]. Diverse IoT scenarios impose varying requirements for authentication mechanisms. For instance, in healthcare IoT systems, device authentication must align with stringent privacy regulations like HIPAA to safeguard patient data. At the same time, industrial IoT environments demand real-time authentication to secure machinery and prevent unauthorized access. Similarly, smart city applications often require scalable identity management to handle dynamic additions and removals of devices such as connected traffic lights and environmental sensors [28–30].

One of the primary challenges in IoT device authentication is establishing trust in environments where devices are often deployed in untrusted or hostile settings [31]. Identity management in IoT extends beyond mere authentication to encompass the entire lifecycle of device identities, from provisioning and registration to revocation and decommissioning. One of the key challenges in IoT identity management is the dynamic nature of IoT environments, where devices frequently join and leave the network, often with little or no human intervention. This dynamic nature requires identity management systems that can operate autonomously, ensuring that devices are securely integrated into the network and that their identities can be managed without compromising security [32–36].

Traditional authentication mechanisms, such as Public Key Infrastructure (PKI), though widely used in conventional IT systems, are often impractical for IoT devices due to their resource-intensive nature [37–39]. IoT devices, particularly those with limited processing power and memory, cannot easily handle the computational overhead associated with PKI's key generation, encryption, and decryption processes. This limitation has spurred the development of alternative authentication methods specifically tailored to the constraints of IoT devices [40].

Elliptic Curve Cryptography (ECC) has emerged as a prominent solution for lightweight authentication in IoT [41,42]. ECC offers comparable security to traditional Rivest–Shamir–Adleman (RSA) encryption but with significantly smaller key sizes, leading to reduced computational and storage requirements [43]. The reduced key size translates into faster processing times and lower energy consumption, making ECC particularly well suited for IoT devices that operate on limited power sources [44]. However, the implementation of ECC in IoT environments requires the careful consideration of various factors such as key management, resistance to side-channel attacks, and the ability to scale across large networks [45].

In addition to ECC, symmetric key cryptography remains a popular choice for IoT authentication due to its efficiency in terms of both speed and energy consumption [46]. Protocols like the Advanced Encryption Standard (AES) are frequently employed for device authentication in IoT systems [47,48]. The AES, with key lengths of 128, 192, and 256 bits, is widely used for device authentication in IoT systems. A 128-bit key provides sufficient security against brute-force attacks while minimizing computational overhead, making it suitable for resource-constrained devices. Longer keys, such as 256-bit, offer enhanced security but increase processing requirements. The secure exchange of AES keys, often facilitated by the Diffie–Hellman or Elliptic Curve Diffie–Hellman protocols, can introduce vulnerabilities like susceptibility to man-in-the-middle attacks if not implemented correctly [49]. While AES ensures strong encryption and efficiency, its reliance on secure

key management remains a limitation in large-scale IoT networks. However, the challenge with symmetric key cryptography lies in the secure distribution and management of keys, particularly in large-scale IoT deployments [50]. Secure key distribution protocols, such as the Diffie–Hellman key exchange, have been adapted for IoT but these still face challenges related to scalability and the potential for man-in-the-middle attacks during the key exchange process [51–53].

Beyond cryptographic techniques, physical unclonable functions (PUFs) have gained attention as a novel approach to device authentication in IoT [54,55]. PUFs exploit the inherent manufacturing variations in electronic circuits to generate unique, device-specific responses that can be used for authentication purposes. Since these responses are derived from the physical characteristics of the device, they are difficult to replicate or forge, providing a high level of security. PUF-based authentication is particularly appealing for IoT devices due to its low computational requirements and resistance to cloning attacks. PUFs generate unique responses based on inherent device variations, requiring minimal hardware operations like delay-based measurements. For example, ring oscillator PUFs consume a few hundred nanowatts of power and require only kilobytes of memory for challenge-response storage, making them suitable for resource-constrained IoT devices. These architectures are integrated into MCUs and FPGAs for lightweight authentication, with error-correction techniques ensuring reliability under environmental variations [56,57]. While PUF-based authentication is highly effective, deploying it at scale presents challenges, such as the need for standardized interfaces and protocols to integrate PUFs into diverse IoT ecosystems [58–60].

Decentralized identity management systems, leveraging blockchain and distributed ledger technologies (DLTs), have been proposed as a solution to the challenges of scalability and trust in IoT networks. By using blockchain, IoT devices can register their identities in a tamper-proof ledger that is distributed across the network. This approach provides several advantages, including the elimination of single points of failure and the ability to audit identity transactions transparently [61–63]. Smart contracts, which are self-executing contracts with the terms of the agreement directly written into code, can be utilized within blockchain-based identity systems to automate identity verification processes and enforce access control policies [64]. However, the integration of blockchain with IoT also raises issues related to the overhead of consensus mechanisms, the latency of blockchain transactions, and the energy consumption associated with maintaining a distributed ledger [65].

Another emerging trend in IoT identity management is the use of federated identity models, which allow IoT devices to share identity credentials across multiple domains or organizations. Federated identity management can simplify the process of integrating IoT devices into diverse and multi-stakeholder environments, such as smart cities or industrial IoT systems, by enabling devices to authenticate across different networks without needing to re-establish their identities. This model, however, requires robust trust frameworks and interoperability standards to ensure that identity information can be securely shared and managed across different domains [66–69].

Identity management also plays a crucial role in ensuring the secure decommissioning of IoT devices. When a device is retired or no longer trusted, its identity must be revoked to prevent it from being used as a vector for attacks. This process, known as identity revocation, must be carried out securely and efficiently to ensure that compromised or outdated devices do not remain a threat to the network [70]. Techniques such as certificate revocation lists (CRLs) and the Online Certificate Status Protocol (OCSP) have been adapted for IoT environments, but these solutions often face challenges related to the timely propagation of revocation information and the overhead of maintaining revocation infrastructure [71–74].

In summary, device authentication and identity management in IoT are complex, multifaceted challenges that require solutions tailored to the unique characteristics of IoT environments. The development of lightweight, scalable, and secure authentication mechanisms is critical to ensuring the integrity and trustworthiness of IoT networks. As the IoT continues to evolve, the integration of emerging technologies such as PUFs, blockchain, and federated identity models will be essential in addressing the security challenges associated with the massive scale and heterogeneity of IoT devices [75–77]. A summary of the works that analyzed the aspects of device authentication and identity management in IoT is captured in Table 2.

Table 2. Summary of studies related to device authentication and identity management in IoT.

Topic	References	Summary
IoT Device Authentication	[26–36]	IoT device authentication and identity management are crucial for securing heterogeneous and dynamic IoT environments, addressing challenges such as trust, scalability, and compliance with specific requirements across diverse applications like healthcare, industrial, and smart city systems.
PKI	[37–40]	Focuses on the challenges and applications of PKI in IoT, including the computational overhead and the specific issues related to managing and distributing keys in IoT systems.
ECC	[41–45]	Examines ECC as an efficient cryptographic method for IoT devices, offering robust security with smaller key sizes that are suitable for devices with limited processing power.
Symmetric Key Cryptography	[46–53]	Discusses the use of symmetric key cryptography, particularly AES, for IoT authentication. It includes challenges related to secure key management, such as the adaptation of Diffie–Hellman key exchange, and addresses issues of scalability and security in large IoT networks.
PUFs	[54–60]	Covers PUFs as a hardware-based authentication method in IoT, leveraging unique physical characteristics of devices to provide secure and low-cost authentication solutions. These references discuss implementation challenges and potential applications in diverse IoT environments.
Decentralized and Blockchain-Based Identity Management	[61–65]	Discusses the role of blockchain and decentralized ledger technologies in managing IoT identities. These references highlight the potential of blockchain to provide secure, scalable, and tamper-proof identity management systems in IoT.
Federated Identity Models	[66–69]	Focuses on federated identity management in IoT, enabling devices to authenticate across multiple domains or networks without needing to re-establish their identities. The references discuss trust frameworks and the integration of federated identity systems in complex IoT environments.
Identity Revocation	[70–77]	These references address the processes for revoking the identities of IoT devices that are no longer trusted or operational. Methods like CRLs, OCSP, and blockchain-based revocation systems are discussed.

3. Data Integrity and Privacy

The challenges of data integrity and privacy in the IoT ecosystem are both profound and multifaceted, driven by the inherent characteristics of IoT networks, including heterogeneity, scale, and the frequent deployment of devices in untrusted or physically exposed environments. Ensuring data integrity within this context requires more than the application of conventional cryptographic techniques [78]; it demands a nuanced understanding of the specific threats and the development of solutions that are adaptable to the constraints and operational contexts of IoT devices [79].

3.1. Advanced Cryptographic Techniques for Data Integrity

Traditional methods such as hash functions [80–82], digital signatures [83–85], and message authentication codes (MACs) [86–88] are commonly utilized to ensure data integrity by verifying that data have not been altered. However, the resource constraints of IoT devices necessitate the adaptation of these techniques. Lightweight cryptographic algorithms like lightweight block ciphers (e.g., PRESENT, SPECK) and reduced-round hash functions have been specifically designed to operate within the limited computational capabilities of IoT devices [89–91]. Despite these adaptations, ensuring the integrity of data in transit and at rest remains challenging, particularly in environments where devices are exposed to physical tampering or where secure key management is problematic [92].

3.1.1. Homomorphic Encryption: Technical Depth and Mathematical Framework

Emerging research has also explored the use of homomorphic encryption, which is a cryptographic technique that allows computations to be carried out directly on encrypted data, producing results that, when decrypted, correspond to those obtained by performing the same operations on the plaintext. This property ensures that data remain confidential during computation, a critical requirement in secure data processing [93,94].

The mathematical foundation of homomorphic encryption can be described as follows. Let $\mathcal{E}$ be the encryption function, $\mathcal{D}$ the decryption function, and $\oplus$ and $\otimes$ the addition and multiplication operations, respectively. For a plaintext space $\mathcal{P}$ and ciphertext space $\mathcal{C}$, homomorphic encryption satisfies the following properties. For addition, the scheme ensures $\mathcal{E}(m_1) \oplus \mathcal{E}(m_2) = \mathcal{E}(m_1 + m_2)$, where $m_1, m_2 \in \mathcal{P}$. For multiplication, it guarantees $\mathcal{E}(m_1) \otimes \mathcal{E}(m_2) = \mathcal{E}(m_1 \cdot m_2)$. The decryption of the resulting ciphertext yields the corresponding operation performed on the plaintext, such that $\mathcal{D}(\mathcal{E}(m_1) \oplus \mathcal{E}(m_2)) = m_1 + m_2$ and $\mathcal{D}(\mathcal{E}(m_1) \otimes \mathcal{E}(m_2)) = m_1 \cdot m_2$.

Homomorphic encryption can be categorized into three main types based on the extent of operations it supports. Partially homomorphic encryption allows either addition or multiplication but not both. For example, RSA (Rivest–Shamir–Adleman) is a partially homomorphic encryption scheme supporting multiplication. In RSA, encryption is performed as $\mathcal{E}(m) = m^e \mod N$, where e is the public exponent and N is the modulus. For two plaintexts m_1 and m_2, the multiplicative homomorphic property ensures $\mathcal{E}(m_1) \cdot \mathcal{E}(m_2) \mod N = (m_1 \cdot m_2)^e \mod N$, preserving the multiplication under encryption [95].

Somewhat homomorphic encryption extends partially homomorphic encryption by supporting a limited number of additions and multiplications. This approach is exemplified by schemes such as Brakerski–Gentry–Vaikuntanathan, which use polynomial rings and modular arithmetic to enable operations while controlling computational complexity [96].

Fully homomorphic encryption represents the most powerful form of homomorphic encryption, allowing an arbitrary number of additions and multiplications on ciphertexts. A foundational fully homomorphic encryption scheme, such as Gentry's construction, builds upon the learning-with-errors problem, a lattice-based cryptographic challenge believed to be resistant to quantum attacks. The key generation process involves selecting a secret key $\mathbf{s} \in \mathbb{Z}_q^n$, a public key matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, and a noise vector $\mathbf{e} \in \mathbb{Z}_q^m$. The ciphertext is formed as $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \mod q$.

Encryption in fully homomorphic encryption involves encoding the plaintext $m \in \{0, 1\}$ by adding it to the ciphertext modulus q scaled by $\lfloor q/2 \rfloor$. Thus, the ciphertext is expressed as $\mathcal{E}(m) = (\mathbf{A}, \mathbf{b} + m \cdot \lfloor q/2 \rfloor \mod q)$. Decryption is achieved by recovering m through modular arithmetic using the secret key $\mathbf{s}$. Specifically, the plaintext is obtained as $m = \lceil (\mathbf{b} - \mathbf{A}\mathbf{s})/\lfloor q/2 \rfloor \rceil \mod 2$.

The evaluation of operations on ciphertexts is performed using modular arithmetic. The addition and multiplication of encrypted values increase the noise inherent in ciphertexts. Fully homomorphic encryption schemes address this by periodically reducing noise through bootstrapping, which involves refreshing ciphertexts to enable continued operations. Homomorphic encryption, particularly fully homomorphic encryption, represents a significant advancement in cryptographic techniques by enabling secure computation on encrypted data. Its mathematical rigor ensures the preservation of confidentiality throughout the data lifecycle, providing a robust foundation for privacy-preserving applications [97].

3.1.2. Quantum Computing and IoT Security

Quantum computing represents a transformative technological development, offering unparalleled computational capabilities while simultaneously posing significant threats to the security of classical cryptographic systems. In the context of IoT security, the emergence of quantum computing necessitates a reevaluation of cryptographic protocols to ensure long-term resilience against quantum attacks [98].

Classical cryptographic systems, such as RSA and elliptic curve cryptography, rely on the computational difficulty of problems like integer factorization and the discrete logarithm problem. Quantum algorithms, such as Shor's algorithm, efficiently solve these problems, rendering these cryptosystems vulnerable. For example, RSA encryption relies on the hardness of factoring a large composite modulus N, where $N = p \cdot q$ for two large primes p and q. Shor's algorithm can factor N in polynomial time, undermining the security of RSA-encrypted communications. To mitigate these risks, the field of post-quantum cryptography focuses on developing cryptographic protocols resistant to quantum attacks. These protocols are based on mathematical problems believed to remain intractable even for quantum computers, such as lattice-based cryptography, code-based cryptography, hash-based signatures, and multivariate polynomial problems [99].

Lattice-based cryptography is particularly promising due to its versatility and efficiency. One example is the learning-with-errors problem, which serves as the foundation for many lattice-based schemes. The learning-with-errors problem can be stated as follows. Given a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$, and a noise vector $\mathbf{e} \in \mathbb{Z}_q^m$, compute the vector $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \mod q$. The task of recovering $\mathbf{s}$ from $(\mathbf{A}, \mathbf{b})$ is computationally hard, even with quantum resources. This hardness forms the basis of several cryptographic primitives, including encryption schemes, key exchanges, and digital signatures [100].

Code-based cryptography leverages the hardness of decoding random linear codes. One notable example is the McEliece cryptosystem, which uses error-correcting codes to provide secure communication. Similarly, hash-based signatures, such as those based on Merkle trees, offer quantum-resistant authentication mechanisms by relying on the one-way nature of cryptographic hash functions [101].

Multivariate polynomial cryptography is another approach where the security relies on the difficulty of solving systems of multivariate quadratic equations over finite fields. Protocols based on this approach are efficient and suitable for resource-constrained environments, such as IoT devices. The adoption of post-quantum cryptographic protocols presents unique challenges for IoT systems. Many IoT devices are resource-constrained, with limited computational power, memory, and energy availability. Implementing quantum-resistant algorithms in such environments requires careful optimization to balance security and performance. Additionally, the transition to post-quantum cryptography involves ensuring compatibility with existing systems and standards, which adds complexity to the integration process [102].

3.2. Data Integrity in Distributed Architectures

IoT systems often rely on distributed architectures, such as edge computing, where data processing occurs at the edge of the network, closer to the data source. In such architectures, ensuring data integrity requires secure communication channels, resilient storage mechanisms, and robust synchronization protocols between edge devices and the central cloud [103,104]. Techniques such as secure multiparty computation (SMPC) and threshold cryptography have been explored to enable secure data aggregation and processing across distributed nodes, ensuring that even if some nodes are compromised, the overall integrity of the data is maintained [105–108].

Blockchain technology has also been proposed as a means to enhance data integrity in distributed IoT systems. By creating an immutable and distributed ledger of transactions, blockchain can ensure that data records are tamper-proof and verifiable by all participants in the network. However, the integration of blockchain into IoT presents significant challenges, particularly in terms of scalability and energy efficiency. The high computational cost and latency associated with consensus mechanisms like proof of work make blockchain less suited for real-time IoT applications, driving the need for alternative consensus algorithms, such as proof-of-stake or directed acyclic graphs (DAGs), which are more resource-efficient [109–114].

Secure Multiparty Computation and Threshold Cryptography

SMPC and threshold cryptography are advanced cryptographic techniques designed to enhance the security and privacy of distributed systems. These methods are particularly relevant in scenarios involving sensitive data processing across multiple entities, such as in IoT environments, where devices often interact within decentralized and untrusted networks. SMPC enables a group of parties to jointly compute a function over their inputs while keeping those inputs private. This ensures that no individual party learns the inputs of the others beyond what is revealed by the function's output. Formally, consider n parties, each with a private input x_i, and a function $f(x_1, x_2, \ldots, x_n)$ to be computed collaboratively. SMPC ensures that the parties compute the function correctly while preserving the privacy of x_i for all i.

The mathematical framework of SMPC typically relies on secret sharing schemes [115]. In Shamir's secret sharing, for instance, a secret s is divided into n shares $(s_1, s_2, \ldots, s_n)$ such that any subset of t or more shares can reconstruct the secret, while fewer than t shares reveal no information about s [116]. This is achieved using polynomial interpolation. The secret s is represented as the constant term of a randomly chosen polynomial $P(x)$ of degree $t-1$ over a finite field $\mathbb{F}_q$:

$$P(x) = s + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1},$$

where $a_1, a_2, \ldots, a_{t-1}$ are random coefficients. Each party is given a share $s_i = P(x_i)$, where x_i is a unique identifier for that party.

The privacy and correctness of SMPC are ensured through protocols such as the GMW (Goldreich–Micali–Wigderson) and BGW (Ben–Or–Goldwasser–Wigderson) protocols. These protocols allow secure computation of any function represented as a Boolean or arithmetic circuit by ensuring that intermediate values during computation remain secret [117].

Threshold cryptography extends the principles of secret sharing to cryptographic operations, enabling secure key management and distributed trust [118]. In a (t, n)-threshold scheme, a private cryptographic key k is divided into n shares, such that any t or more shares can reconstruct k, but fewer than t shares reveal no information about the key.

Threshold cryptography is particularly useful in distributed systems for tasks such as digital signing, decryption, and key generation. For example, in a threshold digital signature scheme, a group of parties collectively signs a message without reconstructing the private signing key. Let $H(m)$ represent the hash of a message m, and let the private key k be shared among n parties. A subset of t parties collaborates to produce partial signatures, which are then combined into a valid signature on m. Mathematically, threshold cryptography leverages techniques such as Lagrange interpolation for share reconstruction. Given t shares $(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t)$, the secret k can be reconstructed as

$$k = \sum_{i=1}^{t} y_i \prod_{j \neq i} \frac{x_j}{x_j - x_i} \mod q,$$

where q is the modulus of the finite field.

3.3. Privacy Preservation in IoT

Privacy concerns in IoT extend beyond the mere protection of data during transmission and storage; they encompass the broader issue of preventing the unauthorized inference of sensitive information. IoT devices often generate vast amounts of data that, when aggregated and analyzed, can reveal patterns and insights about individuals or organizations. This has given rise to the concept of data minimization, where only a minimal amount of necessary data are collected and processed, thus reducing the risk of privacy breaches [119–121].

Techniques such as differential privacy have gained traction as a means to balance data utility with privacy. Differential privacy is a robust mathematical framework designed to protect individual privacy during data analysis. By introducing controlled noise into computations, differential privacy ensures that the inclusion or exclusion of a single data point has a negligible impact on the output, thereby safeguarding sensitive information about individuals [122–124].

Differential privacy is formally defined as follows. Let $\mathcal{M}$ be a randomized algorithm that operates on a dataset D, producing an output in a range $\mathcal{R}$. The algorithm $\mathcal{M}$ satisfies ϵ-differential privacy if, for all datasets D and D' differing in at most one element, and for all subsets $\mathcal{S} \subseteq \mathcal{R}$,

$$\Pr[\mathcal{M}(D) \in \mathcal{S}] \leq e^{\epsilon} \cdot \Pr[\mathcal{M}(D') \in \mathcal{S}],$$

where $\epsilon > 0$ is the privacy loss parameter. A smaller ϵ indicates stronger privacy guarantees.

This definition ensures that an observer cannot confidently determine whether any individual's data are included in the dataset based on the output of $\mathcal{M}$. The parameter ϵ controls the trade-off between privacy and utility: lower values of ϵ provide stronger privacy but may reduce the utility of the output.

Differential privacy is typically achieved by adding noise to the output of a computation. Two commonly used mechanisms are the Laplace Mechanism and the Gaussian Mechanism [125,126].

Laplace Mechanism:

For a function $f : D \to \mathbb{R}^k$, the Laplace Mechanism adds noise sampled from the Laplace distribution. If Δf is the global sensitivity of f, defined as

$$\Delta f = \max_{D,D'} \|f(D) - f(D')\|_1,$$

the mechanism outputs

$$\mathcal{M}(D) = f(D) + \text{Lap}(\Delta f / \epsilon),$$

where $\text{Lap}(b)$ denotes the Laplace distribution with scale parameter b.

Gaussian Mechanism:

The Gaussian Mechanism is suitable when privacy is measured using (ϵ, δ)-differential privacy, a relaxed version of the standard definition. It adds Gaussian noise with variance proportional to the global sensitivity:

$$\mathcal{M}(D) = f(D) + \mathcal{N}(0, \sigma^2), \sigma = \sqrt{2\ln(1.25/\delta)} \cdot \Delta f/\epsilon.$$

In addition to differential privacy, the concept of federated learning has emerged as a promising approach to privacy-preserving ML in IoT. In federated learning, instead of sending raw data to a central server, IoT devices locally train ML models on their data and then send only the model updates to the central server. This approach significantly reduces the amount of sensitive data that need to be transmitted and stored centrally, thus enhancing privacy. However, federated learning introduces new challenges, such as ensuring the integrity and authenticity of the model updates and mitigating the risk of model inversion attacks, where adversaries attempt to reconstruct the original data from the model updates [127–129].

3.4. Privacy in Resource-Constrained Devices

The resource limitations of many IoT devices pose significant challenges for implementing robust privacy-preserving mechanisms. Traditional encryption methods, while effective, are often too resource-intensive for small, battery-powered devices [130]. To address this, lightweight encryption algorithms, such as the Advanced Encryption Standard (AES) in its truncated versions or custom lightweight block ciphers, have been developed to provide a balance between security and performance [131–134].

Moreover, the adoption of hardware-based security features, such as Trusted Execution Environments (TEEs) and PUFs, has been explored as a means to enhance privacy in resource-constrained IoT devices. TEEs provide a secure area within the device's processor that can execute code and store data in a manner that is isolated from the rest of the system, thereby protecting sensitive information even if the device is compromised [135–137]. PUFs, on the other hand, exploit the unique physical characteristics of semiconductor devices to generate unique cryptographic keys that cannot be replicated, thus providing a hardware-based root of trust [138–140].

3.5. Emerging Threats and Countermeasures

As the IoT continues to evolve, so do the threats to data integrity and privacy. One emerging threat is the use of AI by adversaries to carry out sophisticated attacks, such as data poisoning or model inversion, which can compromise both the integrity and privacy of IoT systems. Countering these threats requires a multi-faceted approach, combining traditional cybersecurity measures with advanced AI-based defenses, such as anomaly detection systems that can identify and mitigate suspicious activities in real-time [141–143].

In conclusion, ensuring data integrity and privacy in IoT systems is a complex and ongoing challenge that requires the continuous development and refinement of security mechanisms. As IoT continues to permeate every aspect of our lives, the importance of robust data integrity and privacy protection cannot be overstated. Thus, we should focus on developing scalable, efficient, and adaptable security solutions that can keep pace with the rapidly evolving IoT landscape, ensuring that the benefits of IoT can be realized without compromising security [144,145]. A summary of the works that analyzed the aspects of data integrity and privacy in IoT is shown in Table 3.

Table 3. A summary of studies related to data integrity and privacy in the IoT.

Topic	References	Summary
Advanced Cryptographic Techniques for Data Integrity	[79–102]	Discusses traditional and lightweight cryptographic techniques such as hash functions, digital signatures, and homomorphic encryption. Also covers post-quantum cryptography and its applicability to IoT data integrity.
Data Integrity in Distributed Architectures	[103–118]	Explores methods like SMPC, threshold cryptography, and blockchain technology for maintaining data integrity in distributed IoT systems, particularly in edge computing environments.
Privacy Preservation in IoT	[119–129]	Focuses on privacy-preserving techniques such as differential privacy and federated learning. Discusses the challenges of implementing these techniques in IoT, especially in scenarios with large-scale data analytics.
Privacy in Resource-Constrained Devices	[130–140]	Analyzes the implementation of privacy-preserving mechanisms in IoT devices with limited resources, including lightweight encryption algorithms and hardware-based security features like TEEs and PUFs.
Emerging Threats and Countermeasures	[141–145]	Identifies emerging threats to data integrity and privacy in IoT, such as AI-based attacks, and discusses potential countermeasures, including AI-driven security solutions and anomaly detection systems.

4. Network Security and Communication Protocols

The security of IoT networks is a complex and multifaceted challenge, deeply rooted in the diversity and heterogeneity of the communication protocols that facilitate interactions among IoT devices. These protocols, which govern the data exchange processes, are critical to ensuring the confidentiality, integrity, and availability of information transmitted across the IoT ecosystem. Given the limited resources of IoT devices, protocols are often designed with an emphasis on efficiency, which can result in trade-offs with security. This section delves into the intricacies of IoT network security, exploring the specific vulnerabilities associated with widely used communication protocols, advanced mitigation techniques, and emerging trends in protocol design and security [146–148].

4.1. Security Challenges in IoT Protocols

IoT communication protocols are essential for enabling seamless connectivity and data exchange in IoT ecosystems. However, their design for efficiency and resource-constrained environments often leads to significant security challenges. This section discusses the security issues associated with widely used protocols, including MQTT (Message Queuing Telemetry Transport) [149–151], CoAP (Constrained Application Protocol) [152–154], Zigbee [155–157], and additional protocols such as Bluetooth [158], Z-Wave [159], LoRaWAN [160], Thread [161], and Wi-Fi [162].

MQTT, which is widely used in industrial IoT and smart home applications, operates on a publish–subscribe model that, while efficient, is susceptible to several attack vectors, including man-in-the-middle (MitM) attacks [163], replay attacks [164], and unauthorized access due to its default lack of encryption [165]. Although TLS (Transport Layer Security) can be layered over MQTT to secure communication, the overhead introduced by TLS can be prohibitive for resource-constrained IoT devices [166].

CoAP, designed for use in lightweight M2M (machine-to-machine) communication, is another protocol that faces security issues due to its reliance on UDP (User Datagram Protocol), which inherently lacks the reliability and security features of TCP (Transmission

Control Protocol) [167]. CoAP's use of DTLS (Datagram Transport Layer Security) to secure its communications adds a layer of protection, but challenges such as the proper implementation of DTLS in constrained environments and the potential for DoS attacks remain significant concerns [168,169].

Zigbee, commonly used in wireless sensor networks and smart home applications, operates on the IEEE 802.15.4 standard and uses AES-128 for encryption. Despite this, Zigbee networks have been shown to be vulnerable to attacks such as key extraction, traffic analysis, and device impersonation. The open nature of the Zigbee protocol stack, while beneficial for interoperability, also exposes it to a wide range of potential security threats [170–172].

Bluetooth, particularly Bluetooth Low Energy (BLE), is widely used in wearable devices and short-range IoT systems. Although BLE improves energy efficiency, it remains vulnerable to attacks like eavesdropping, replay attacks, and unauthorized device pairing. Enhanced authentication mechanisms and the adoption of Secure Simple Pairing (SSP) protocols can improve security in Bluetooth-based IoT applications [173,174].

Z-Wave, a low-power communication protocol often used in home automation systems, faces threats such as DoS attacks and encryption key compromise. Strengthening Z-Wave networks requires improved key management systems, regular firmware updates, and the integration of advanced encryption protocols [175,176].

LoRaWAN, a protocol for long-range, low-power IoT communication, is commonly used in smart agriculture, logistics, and urban infrastructure. It faces challenges like session key distribution vulnerabilities and replay attacks. Adopting end-to-end encryption and secure key provisioning methods can help secure LoRaWAN-based systems [177,178].

Thread, an IPv6-based low-power protocol for connected devices, offers strong security through its mesh networking capabilities. However, effective key management remains critical to preventing unauthorized network access and data breaches [179,180].

Wi-Fi, while ubiquitous in IoT applications, has its own vulnerabilities, such as susceptibility to key reinstallation attacks (e.g., KRACK) and weak password policies. Advanced security protocols like Wi-Fi Protected Access 3 (WPA3) provide enhanced encryption and authentication mechanisms, addressing many traditional Wi-Fi security issues [181,182].

4.2. Advanced Security Mechanisms

To address these vulnerabilities, several advanced security mechanisms have been developed and proposed. For MQTT, techniques such as payload encryption, token-based authentication, and the integration of zero-trust architectures are being explored to enhance security without significantly impacting performance. The concept of zero-trust, where every device is authenticated and authorized before it can communicate, is particularly relevant for IoT, where devices may be deployed in potentially hostile environments [183–185].

In CoAP, the use of object security, where security is applied directly to the data being transmitted rather than the transport layer, has been proposed as a means to improve security without the overhead associated with DTLS [186]. This approach, combined with the implementation of secure key management protocols such as OSCORE (Object Security for Constrained RESTful Environments), can mitigate many of the security risks inherent in CoAP-based systems [187,188].

Zigbee's security can be enhanced through the implementation of stronger key management practices and the use of frequency hopping techniques to make it more difficult for attackers to intercept and analyze communications. Additionally, ongoing research into the use of quantum-resistant cryptography for IoT networks promises to provide long-term

security solutions that can withstand the advances in computational power expected in the coming years [189–191].

4.3. Emerging Trends and Protocol Innovations

As the IoT landscape continues to evolve, new communication protocols and security paradigms are emerging. One significant trend is the integration of Software-Defined Networking (SDN) with IoT, where SDN's centralized control model can provide greater visibility and control over network traffic, allowing for more dynamic and responsive security measures. By decoupling the control and data planes, SDN enables the real-time monitoring of network conditions and the application of security policies that can adapt to emerging threats [192–195].

Another emerging trend is the use of blockchain technology to enhance the security and integrity of IoT networks. Blockchain's decentralized and tamper-resistant nature makes it an attractive option for managing IoT device identities and securing communications. Protocols such as IOTA, designed specifically for IoT, leverage the Tangle architecture—a DAG that allows for scalable and secure transactions without the need for miners, which is resource-intensive. This approach is particularly suited for IoT environments, where resources are constrained and traditional blockchain solutions may be impractical [196–198].

Additionally, the development of protocol-agnostic security frameworks is gaining traction, where security measures are abstracted from the specific protocol and instead applied at a higher layer of the network stack. This allows for a more uniform application of security policies across heterogeneous networks, addressing one of the key challenges in IoT network security [199–201].

A summary of the above works that analyze network security and communication protocols is captured in Table 4.

Table 4. A list of topics and related studies dedicated to network security and communication protocols.

Topic	References	Summary
Communication Protocols Overview	[146–148]	Surveys and reviews on IoT communication protocols and their performance.
MQTT Protocol	[149–151]	Studies on the MQTT protocol, including use cases, security challenges, and implementations.
CoAP Protocol	[152–154]	Research on the CoAP protocol, focusing on securing communications and performance analysis.
Zigbee Protocol	[155–157]	Analysis of the Zigbee protocol, including its application in various networks and security aspects.
Bluetooth Protocol	[158,173,174]	Overview of Bluetooth vulnerabilities, such as eavesdropping, replay attacks, and pairing security issues.

Table 4. *Cont.*

Topic	References	Summary
Z-Wave Protocol	[159,175,176]	Security analysis of Z-Wave, including challenges like DoS attacks and encryption key compromise.
LoRaWAN Protocol	[160,177,178]	Examines LoRaWAN security challenges, such as session key vulnerabilities and replay attacks.
Thread Protocol	[161,179,180]	Studies on the Thread protocol, highlighting security through mesh networking and key management.
Wi-Fi Security Issues	[162,181,182]	Discusses Wi-Fi vulnerabilities such as KRACK attacks and advancements with WPA3 encryption.
MQTT Security Issues	[163–166]	Discussions on security challenges in MQTT, including attacks like MitM and DoS.
CoAP Security Enhancements	[167–169]	Enhancements to CoAP security, including DTLS vulnerabilities and protection methods.
Zigbee Security Vulnerabilities	[170–172]	Exploration of security vulnerabilities in Zigbee networks, such as key extraction and device impersonation.
Advanced MQTT Security Mechanisms	[183–185]	Proposed enhancements to MQTT security through payload encryption and zero-trust architecture.
Advanced CoAP Security Mechanisms	[186–188]	Improvement strategies for CoAP security, such as OSCORE for better key management.
Zigbee Quantum-Resistant Cryptography	[189–191]	Studies on the application of quantum-resistant cryptography in Zigbee networks.
SDN Integration with IoT	[192–195]	Research on integrating SDN with IoT for improved security.
Blockchain and IoT	[196–198]	Use of blockchain technology in IoT to enhance security and integrity.
Protocol-Agnostic Security	[199–201]	Development of security frameworks that are abstracted from specific communication protocols.

5. Artificial Intelligence and Machine Learning in IoT Security

The integration of AI and ML into IoT security represents a paradigm shift, offering both novel defense mechanisms and enhancing existing cybersecurity frameworks. In the context of IoT, AI and ML are not merely tools for automation but are fundamental in addressing the unique security challenges posed by the complexity, scale, and dynamic nature of IoT environments [202–204].

One of the most critical applications of AI in IoT security is anomaly detection [205]. Traditional security systems rely on predefined rules and signatures to identify threats [206]. However, IoT networks are characterized by high variability in device behavior, making it difficult to distinguish between normal and malicious activities using static rules. ML

algorithms, particularly those employing unsupervised learning, can be trained to recognize patterns in network traffic and device behavior, enabling the detection of anomalies that may indicate security breaches [207]. Techniques such as clustering, autoencoders, and generative adversarial networks (GANs) are increasingly being employed to identify deviations from normal behavior, which can be indicative of a cyberattack [208–210].

Moreover, supervised learning models are being utilized to enhance IDS within IoT networks. By training on large datasets that include labeled instances of both normal and malicious activities, these models can classify incoming traffic in real time, improving the accuracy and speed of threat detection [211–213]. Ensemble methods, which combine the strengths of multiple learning algorithms, have shown particular promise in boosting detection rates while minimizing false positives. Furthermore, the deployment of federated learning techniques allows models to be trained across multiple IoT devices without the need for centralized data collection, preserving privacy while enhancing security [214–216].

AI and ML also play a crucial role in threat intelligence and predictive analytics within IoT security [217]. The vast amount of data generated by IoT devices can be leveraged to identify emerging threats and predict potential security incidents. Natural language processing (NLP) techniques are used to analyze unstructured data from threat reports, social media, and dark web forums, extracting actionable insights that can inform proactive defense strategies. Additionally, time-series forecasting models can be applied to predict the occurrence of security events based on historical data, enabling the development of predictive maintenance schedules and proactive security measures [218–221].

Another significant application is in the domain of secure device management. AI-driven solutions are being developed to automate the process of device enrollment, configuration, and firmware updates in IoT ecosystems. These systems can identify vulnerabilities in device firmware and initiate automated updates, reducing the window of exposure to potential exploits [222–225]. Furthermore, AI can enhance the security of device authentication processes by incorporating behavioral biometrics and continuous authentication mechanisms. These approaches analyze patterns in device usage and interactions, allowing for the dynamic assessment of trust and the detection of compromised devices [226–228].

In the context of cryptographic security, AI and ML are being employed to develop more resilient encryption schemes tailored to the resource constraints of IoT devices. For instance, AI-driven optimization techniques can be used to design lightweight cryptographic algorithms that balance security with computational efficiency [229–231]. Additionally, AI is being explored as a means to secure cryptographic keys through techniques such as quantum key distribution (QKD), where ML models are used to optimize key generation and distribution processes, enhancing the security of communication channels in IoT networks [232–234].

However, the integration of AI into IoT security is not without challenges. Adversarial ML is an emerging threat where attackers craft inputs specifically designed to deceive AI models. In the context of IoT, adversarial attacks can be particularly damaging, as they can lead to the misclassification of malicious activities as benign or vice versa. Research into robust AI models that can withstand such adversarial manipulations is critical. Techniques such as adversarial training, where models are trained on both legitimate and adversarial examples, are being explored to enhance the resilience of AI systems in IoT environments [235–240].

Additionally, the use of AI in IoT security raises concerns about the interpretability and transparency of ML models. In critical applications such as healthcare or industrial automation, it is essential not only to detect security breaches but also to understand the reasoning behind the AI's decisions. Explainable AI (XAI) is a burgeoning field that seeks

to make ML models more transparent, allowing security analysts to comprehend the basis for a model's predictions and to trust its outputs in high-stakes scenarios [241–245].

The deployment of AI in IoT security also intersects with issues of scalability and energy efficiency. IoT networks can comprise millions of devices, each generating data that need to be processed in real time. Scaling AI solutions to handle this data deluge without overwhelming the computational resources of IoT devices is a significant challenge. Edge AI, where ML models are deployed directly on IoT devices or nearby edge servers, is emerging as a solution to this challenge. By processing data closer to the source, edge AI reduces latency and bandwidth usage while maintaining robust security measures. Research into low-power AI models, which can function within the strict energy constraints of IoT devices, is also gaining traction, with techniques such as model pruning and quantization being used to optimize the energy efficiency of AI algorithms [246–250]. The previously discussed works that describe the synergy of AI and ML for cybersecurity in IoT are listed in Table 5.

Table 5. A summary of studies related to AI and ML in IoT security.

Topic	References	Summary
AI and ML for Anomaly Detection	[202–210]	These references discuss the use of AI and ML techniques, especially unsupervised learning, to detect anomalies in IoT networks.
Supervised Learning for Intrusion Detection	[211–216]	Focuses on supervised learning models to enhance IDS in IoT environments.
AI in Threat Intelligence and Predictive Analytics	[217–221]	References cover AI and ML applications in threat intelligence and predictive analytics for IoT security.
Secure Device Management with AI	[222–228]	These studies explore AI-driven solutions for secure device management, including automated updates and continuous authentication.
AI for Cryptographic Security	[229–234]	Focus on AI applications in developing resilient cryptographic schemes and optimizing key distribution in IoT.
Challenges with Adversarial ML	[235–240]	Discusses the risks and countermeasures associated with adversarial machine learning in IoT security contexts.
XAI for IoT Security	[241–245]	References that delve into the importance of explainability in AI models used for IoT security.
Scalability and Energy Efficiency in AI for IoT	[246–250]	Covers research on scaling AI solutions and improving energy efficiency in IoT, including edge AI techniques.

6. Challenges and Future Directions

The field of IoT cybersecurity is confronted with an array of multifaceted challenges, which are exacerbated by the rapid and often haphazard deployment of IoT devices across diverse environments. These challenges are not only technical in nature but also span regulatory, economic, and operational domains, each with its own set of complexities that must be navigated to secure the IoT ecosystem effectively [251–253].

One of the most pressing challenges is the heterogeneity of IoT devices, which vary widely in terms of hardware capabilities, communication protocols, and security requirements. This diversity is particularly evident across application domains. Industrial IoT systems require real-time protection mechanisms to secure critical operational data and prevent downtime in predictive maintenance processes [254]. Wearable devices, used

extensively in fitness and healthcare, demand advanced privacy mechanisms to safeguard sensitive personal data while respecting user consent [255]. Autonomous vehicles, meanwhile, require ultra-low-latency communication systems to ensure safe navigation and resilience against cyberattacks on sensor data or control systems [256]. These examples highlight the need for domain-specific security strategies that consider the unique operational and regulatory contexts of each IoT application [257–259].

Another significant challenge is the issue of legacy devices, which were not designed with security in mind and are often difficult or impossible to update. These devices represent a substantial portion of the current IoT landscape and pose serious security risks. The presence of these legacy systems creates a situation where new, secure devices must coexist with older, vulnerable ones, leading to a weakest-link problem in the network. Addressing this challenge requires innovative approaches such as retrofitting security mechanisms onto existing devices or developing gateway solutions that can act as a security intermediary between legacy devices and the broader network [260–262].

Interoperability among IoT devices and systems is another critical challenge, particularly in environments where devices from multiple vendors must work together seamlessly. The lack of standardized communication protocols and security frameworks can lead to significant vulnerabilities, as attackers may exploit gaps between different systems. Future research must focus on the development of robust interoperability standards that incorporate security as a foundational element. This includes not only technical standards but also agreements on governance and accountability mechanisms across different stakeholders, ensuring that security responsibilities are clearly defined and enforced [263–265].

Privacy concerns in IoT go beyond the mere protection of data; they encompass the need to provide users with control over their data and transparency about how it is used. The challenge lies in balancing the need for data collection, which is often essential for IoT functionality, with the need to respect user privacy. This is particularly complex in scenarios involving large-scale data aggregation and analysis, where the anonymization techniques may not be sufficient to prevent the re-identification of individuals. Advanced privacy-preserving techniques, such as secure multi-party computation and federated learning, are promising avenues for research, but their integration into IoT systems remains challenging due to resource constraints and the need for real-time processing [266–268].

The dynamic and often unpredictable nature of IoT environments also presents a significant challenge for security. Unlike traditional IT systems, IoT networks are highly distributed and decentralized, with devices frequently joining and leaving the network. This fluidity complicates the implementation of consistent security policies and makes it difficult to detect and respond to threats in real time. Traditional security monitoring tools are often inadequate for IoT environments, necessitating the development of new approaches that can handle the scale and dynamism of IoT networks. Techniques such as distributed ledger technology (DLT) and decentralized identity management systems hold promise in this regard, offering the potential to enhance the security and resilience of IoT networks by distributing trust and reducing single points of failure [269–271].

Emerging technologies such as 5G and edge computing are expected to further complicate the IoT security landscape. While these technologies offer significant performance improvements, they also introduce new attack vectors and exacerbate existing vulnerabilities. For instance, the shift towards edge computing, where data processing occurs closer to the source of data generation, reduces latency but also increases the attack surface by distributing processing across numerous, potentially insecure, edge devices. Securing these edge environments requires new approaches to threat detection and response, including

the use of AI-driven security analytics that can operate in real time and adapt to evolving threats [272–275].

Furthermore, the regulatory environment surrounding IoT security is still in its nascent stages, with many regions lacking comprehensive legal frameworks to address the unique challenges posed by IoT devices. The absence of clear regulations and standards leads to a patchwork of security practices that vary widely across industries and geographies. There is a critical need for international cooperation in establishing regulatory standards that can provide a consistent baseline for IoT security while allowing for regional adaptations. This involves not only technical standards but also frameworks for data protection, privacy, and the ethical use of IoT technologies [276–279].

Economic factors also play a crucial role in shaping the IoT security landscape. The cost of implementing advanced security measures can be prohibitive for many organizations, particularly in sectors such as agriculture and small-scale manufacturing where margins are thin. This economic barrier often leads to security being deprioritized, resulting in vulnerable systems that are ripe for exploitation [280,281].

In conclusion, the challenges facing IoT cybersecurity are vast and varied, requiring a multidisciplinary approach that goes beyond technical solutions to encompass regulatory, economic, and operational considerations. Future research could explore cost-effective security solutions that do not compromise on protection, as well as innovative business models that incentivize investment in IoT security. Also, it should focus on developing adaptive, scalable, and cost-effective security solutions that can keep pace with the rapid evolution of IoT technologies. As the IoT continues to expand, the need for robust, integrated security measures will become increasingly critical, necessitating ongoing collaboration between industry, academia, and government to ensure the safety and resilience of the IoT ecosystem [282–287]. The list of key topics related to challenges of cybersecurity in IoT and future directions is summarized in Table 6.

Table 6. A summary of studies related to challenges and future directions of cybersecurity in the IoT.

Topic	References	Description
Security Challenges in IoT	[251–259]	IoT cybersecurity faces diverse challenges driven by device heterogeneity, regulatory constraints, and application-specific requirements such as real-time protection in IIoT, privacy in wearables, and ultra-low-latency in autonomous vehicles.
Legacy Devices and Security Risks	[260–262]	Focuses on the challenges posed by legacy IoT devices that were not designed with security in mind and potential solutions like retrofitting.
Interoperability and Standardization	[263–265]	Highlights the challenges related to interoperability among IoT devices and the need for standardized communication protocols and security frameworks.
Privacy and Data Protection	[266–268]	Addresses privacy concerns, emphasizing the need for advanced privacy-preserving techniques and secure data management in IoT.
Decentralized Security Solutions	[269–271]	Discusses decentralized approaches like blockchain and DLT for enhancing security in distributed IoT environments.
Emerging Technologies and IoT Security	[272–275]	Explores the impact of emerging technologies like 5G and edge computing on IoT security and the associated new attack vectors.
Regulatory and Economic Considerations	[276–281]	Examines the regulatory landscape and economic factors influencing IoT security, emphasizing the need for international cooperation and cost-effective solutions.
Future Research Directions	[282–287]	Suggests future research directions focusing on scalable security solutions, adaptive strategies, and enhancing stakeholder collaboration for IoT security.

7. Conclusions

The landscape of IoT cybersecurity is marked by both significant advancements and persistent challenges, reflecting the complexity and scale of IoT ecosystems. The diversity of IoT devices, coupled with their widespread deployment across critical sectors, necessitates a multifaceted approach to security that goes beyond traditional measures. The advancements in lightweight cryptographic techniques, privacy-preserving algorithms, and AI-driven anomaly detection represent important strides in securing IoT environments. However, these developments are not sufficient on their own to address the full spectrum of cybersecurity threats faced by IoT systems.

One of the critical insights from this survey is the need for holistic, end-to-end security frameworks that can be seamlessly integrated into the IoT lifecycle—from device manufacturing and deployment to ongoing operation and decommissioning. The lack of standardized security protocols and frameworks across different IoT ecosystems continues to be a significant barrier to achieving robust security. This fragmentation leads to inconsistent security postures across devices and networks, creating vulnerabilities that can be exploited by attackers. Future research must prioritize the development of interoperable and scalable security frameworks that can be adapted to the diverse range of IoT devices and applications.

Moreover, the rapid pace of technological innovation in IoT has outstripped the development of corresponding security measures. As IoT devices become more sophisticated and interconnected, the attack surface grows exponentially, making it increasingly difficult to secure these systems using conventional approaches. This necessitates a shift toward more proactive and adaptive security strategies. The integration of AI and ML in cybersecurity, while promising, also introduces new risks that must be carefully managed. For instance, adversarial attacks on AI models could undermine the effectiveness of these systems, leading to false positives or, worse, undetected breaches. This underscores the importance of developing resilient AI models that can withstand such attacks, as well as the need for continuous monitoring and updating of these models to keep pace with emerging threats.

Another crucial aspect highlighted in this survey is the need for enhanced collaboration between different stakeholders in the IoT ecosystem, including device manufacturers, network operators, service providers, and regulatory bodies. The complexity of IoT security challenges cannot be addressed by any single entity; rather, it requires coordinated efforts across the entire supply chain. This includes the establishment of industry-wide best practices, regulatory standards, and compliance frameworks that ensure a baseline level of security across all IoT devices and networks. The role of government and international regulatory bodies is particularly critical in setting these standards and ensuring their global adoption.

Looking forward, there is a pressing need to focus on the scalability of IoT security solutions. As the number of IoT devices continues to grow exponentially, security mechanisms must be capable of scaling accordingly. This includes not only the scalability of cryptographic and authentication mechanisms but also the ability to manage large-scale, distributed IoT networks securely. Techniques such as DLT, including blockchain, offer promising avenues for achieving decentralized and scalable security in IoT, but their integration into existing IoT infrastructures poses significant challenges that must be overcome.

Finally, the human factor remains a critical, yet often overlooked, component of IoT security. User awareness and education about IoT security risks, as well as the development of user-friendly security interfaces, are essential for ensuring that security measures are

effectively implemented and maintained. As IoT devices increasingly permeate everyday life, the onus is on both manufacturers and users to prioritize security.

In conclusion, while the field of IoT cybersecurity has made considerable progress, it remains a dynamic and rapidly evolving area of research. The complexity of IoT ecosystems, combined with the ever-expanding threat landscape, requires continuous innovation in security approaches. By building on the current advancements and addressing the identified gaps, the research community can develop more robust and adaptive security solutions that are capable of protecting the next generation of IoT devices and networks. The future of IoT security lies in the ability to anticipate and counter emerging threats while ensuring that security solutions are scalable, interoperable, and user-centric.

Author Contributions: E.D. and M.T. conceived of the idea, designed and performed the experiments, analyzed the results, drafted the initial manuscript, and revised the final manuscript. All authors have read and agreed to the published version of the manuscript

Funding: This research received no external funding

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Villamil, S.; Hernández, C.; Tarazona, G. An overview of internet of things. *Telkomnika (Telecommun. Comput. Electron. Control)* **2020**, *18*, 2320–2327. [CrossRef]
2. Greengard, S. *The Internet of Things*; MIT Press: Cambridge, MA, USA, 2021.
3. Malhotra, P.; Singh, Y.; Anand, P.; Bangotra, D.K.; Singh, P.K.; Hong, W.C. Internet of things: Evolution, concerns and security challenges. *Sensors* **2021**, *21*, 1809. [CrossRef] [PubMed]
4. Deep, S.; Zheng, X.; Jolfaei, A.; Yu, D.; Ostovari, P.; Kashif Bashir, A. A survey of security and privacy issues in the Internet of Things from the layered context. *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e3935. [CrossRef]
5. Hathaliya, J.J.; Tanwar, S. An exhaustive survey on security and privacy issues in Healthcare 4.0. *Comput. Commun.* **2020**, *153*, 311–335. [CrossRef]
6. Chanal, P.M.; Kakkasageri, M.S. Security and privacy in IoT: A survey. *Wirel. Pers. Commun.* **2020**, *115*, 1667–1693. [CrossRef]
7. Venu, D.N.; Arun Kumar, A.; Vaigandla, K.K. Review of internet of things (iot) for future generation wireless communications. *Int. J. Mod. Trends Sci. Technol.* **2022**, *8*, 01–08.
8. Gulati, K.; Boddu, R.S.K.; Kapila, D.; Bangare, S.L.; Chandnani, N.; Saravanan, G. A review paper on wireless sensor network techniques in Internet of Things (IoT). *Mater. Today Proc.* **2022**, *51*, 161–165. [CrossRef]
9. Celebi, H.B.; Pitarokoilis, A.; Skoglund, M. Wireless communication for the industrial IoT. In *Industrial IoT: Challenges, Design Principles, Applications, and Security*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 57–94.
10. Hajjaji, Y.; Boulila, W.; Farah, I.R.; Romdhani, I.; Hussain, A. Big data and IoT-based applications in smart environments: A systematic review. *Comput. Sci. Rev.* **2021**, *39*, 100318. [CrossRef]
11. Atiewi, S.; Al-Rahayfeh, A.; Almiani, M.; Yussof, S.; Alfandi, O.; Abugabah, A.; Jararweh, Y. Scalable and secure big data IoT system based on multifactor authentication and lightweight cryptography. *IEEE Access* **2020**, *8*, 113498–113511. [CrossRef]
12. Deepa, N.; Pham, Q.V.; Nguyen, D.C.; Bhattacharya, S.; Prabadevi, B.; Gadekallu, T.R.; Maddikunta, P.K.R.; Fang, F.; Pathirana, P.N. A survey on blockchain for big data: Approaches, opportunities, and future directions. *Future Gener. Comput. Syst.* **2022**, *131*, 209–226. [CrossRef]
13. Karie, N.M.; Sahri, N.M.; Yang, W.; Valli, C.; Kebande, V.R. A review of security standards and frameworks for IoT-based smart environments. *IEEE Access* **2021**, *9*, 121975–121995. [CrossRef]
14. Medhane, D.V.; Sangaiah, A.K.; Hossain, M.S.; Muhammad, G.; Wang, J. Blockchain-enabled distributed security framework for next-generation IoT: An edge cloud and software-defined network-integrated approach. *IEEE Internet Things J.* **2020**, *7*, 6143–6149. [CrossRef]
15. Shah, Z.; Ullah, I.; Li, H.; Levula, A.; Khurshid, K. Blockchain based solutions to mitigate distributed denial of service (DDoS) attacks in the Internet of Things (IoT): A survey. *Sensors* **2022**, *22*, 1094. [CrossRef]
16. Kuzlu, M.; Fair, C.; Guler, O. Role of artificial intelligence in the Internet of Things (IoT) cybersecurity. *Discov. Internet Things* **2021**, *1*, 7. [CrossRef]

17. Lee, I. Internet of Things (IoT) cybersecurity: Literature review and IoT cyber risk management. *Future Internet* **2020**, *12*, 157. [CrossRef]
18. Gimenez-Aguilar, M.; De Fuentes, J.M.; Gonzalez-Manzano, L.; Arroyo, D. Achieving cybersecurity in blockchain-based systems: A survey. *Future Gener. Comput. Syst.* **2021**, *124*, 91–118. [CrossRef]
19. Garrido, C.B.E.; Compte, S.S.; Roldan, L.R.; Malacara, A.A. Survey and testing of the IoT Cybersecurity Framework using intrusion detection systems. *Int. J. Comput. Networks Appl.* **2022**, *9*, 601–623.
20. Kotenko, I.; Izrailov, K.; Buinevich, M. Static analysis of information systems for IoT cyber security: A survey of machine learning approaches. *Sensors* **2022**, *22*, 1335. [CrossRef] [PubMed]
21. Inayat, U.; Zia, M.F.; Mahmood, S.; Khalid, H.M.; Benbouzid, M. Learning-based methods for cyber attacks detection in IoT systems: A survey on methods, analysis, and future prospects. *Electronics* **2022**, *11*, 1502. [CrossRef]
22. Shah, Y.; Sengupta, S. A survey on Classification of Cyber-attacks on IoT and IIoT devices. In Proceedings of the 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 28–31 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 0406–0413.
23. Ashraf, I.; Park, Y.; Hur, S.; Kim, S.W.; Alroobaea, R.; Zikria, Y.B.; Nosheen, S. A survey on cyber security threats in IoT-enabled maritime industry. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 2677–2690. [CrossRef]
24. Matheu, S.N.; Hernandez-Ramos, J.L.; Skarmeta, A.F.; Baldini, G. A survey of cybersecurity certification for the internet of things. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 115. [CrossRef]
25. Ahmad, W.; Rasool, A.; Javed, A.R.; Baker, T.; Jalil, Z. Cyber security in iot-based cloud computing: A comprehensive survey. *Electronics* **2021**, *11*, 16. [CrossRef]
26. Mamdouh, M.; Awad, A.I.; Khalaf, A.A.; Hamed, H.F. Authentication and identity management of IoHT devices: Achievements, challenges, and future directions. *Comput. Secur.* **2021**, *111*, 102491. [CrossRef]
27. Javed, A.; Malhi, A.; Kinnunen, T.; Främling, K. Scalable IoT platform for heterogeneous devices in smart environments. *IEEE Access* **2020**, *8*, 211973–211985. [CrossRef]
28. Tahir, M.; Sardaraz, M.; Muhammad, S.; Saud Khan, M. A lightweight authentication and authorization framework for blockchain-enabled IoT network in health-informatics. *Sustainability* **2020**, *12*, 6960. [CrossRef]
29. Alruwaili, O.; Tanveer, M.; Alotaibi, F.M.; Abdelfattah, W.; Armghan, A.; Alserhani, F.M. Securing the IoT-enabled smart healthcare system: A PUF-based resource-efficient authentication mechanism. *Heliyon* **2024**, *10*, e37577. [CrossRef]
30. Zhang, H.; Babar, M.; Tariq, M.U.; Jan, M.A.; Menon, V.G.; Li, X. SafeCity: Toward safe and secured data management design for IoT-enabled smart city planning. *IEEE Access* **2020**, *8*, 145256–145267. [CrossRef]
31. Alhandi, S.A.; Kamaludin, H.; Alduais, N.A.M. Trust evaluation model in IoT environment: a comprehensive survey. *IEEE Access* **2023**, *11*, 11165–11182. [CrossRef]
32. Rathee, T.; Singh, P. A systematic literature mapping on secure identity management using blockchain technology. *J. King Saud-Univ. Comput. Inf. Sci.* **2022**, *34*, 5782–5796. [CrossRef]
33. Bouras, M.A.; Lu, Q.; Dhelim, S.; Ning, H. A lightweight blockchain-based IoT identity management approach. *Future Internet* **2021**, *13*, 24. [CrossRef]
34. Shobanadevi, A.; Tharewal, S.; Soni, M.; Kumar, D.D.; Khan, I.R.; Kumar, P. Novel identity management system using smart blockchain technology. *Int. J. Syst. Assur. Eng. Manag.* **2022**, *13*, 496–505. [CrossRef]
35. Alamri, B.; Crowley, K.; Richardson, I. Blockchain-based identity management systems in health IoT: A systematic review. *IEEE Access* **2022**, *10*, 59612–59629. [CrossRef]
36. Venkatraman, S.; Parvin, S. Developing an IoT identity management system using blockchain. *Systems* **2022**, *10*, 39. [CrossRef]
37. Mathur, S.; Arora, A. Internet of things (IoT) and PKI-based security architecture. In *Industrial Internet of Things and Cyber-Physical Systems: Transforming the Conventional to Digital*; IGI Global: Hershey, PA, USA, 2020; pp. 25–46.
38. Danquah, P.; Kwabena-Adade, H. Public key infrastructure: an enhanced validation framework. *J. Inf. Secur.* **2020**, *11*, 241–260. [CrossRef]
39. Viriyasitavat, W.; Xu, L.D.; Sapsomboon, A.; Dhiman, G.; Hoonsopon, D. Building trust of Blockchain-based Internet-of-Thing services using public key infrastructure. *Enterp. Inf. Syst.* **2022**, *16*, 2037162. [CrossRef]
40. Höglund, J.; Lindemer, S.; Furuhed, M.; Raza, S. PKI4IoT: Towards public key infrastructure for the Internet of Things. *Comput. Secur.* **2020**, *89*, 101658. [CrossRef]
41. Gulen, U.; Baktir, S. Elliptic curve cryptography for wireless sensor networks using the number theoretic transform. *Sensors* **2020**, *20*, 1507. [CrossRef] [PubMed]
42. Ullah, S.; Zheng, J.; Din, N.; Hussain, M.T.; Ullah, F.; Yousaf, M. Elliptic Curve Cryptography; Applications, challenges, recent advances, and future trends: A comprehensive survey. *Comput. Sci. Rev.* **2023**, *47*, 100530. [CrossRef]

43. Hu, X.; Huang, H.; Zheng, X.; Liu, Y.; Xiong, X. Low-power reconfigurable architecture of elliptic curve cryptography for IoT. *IEICE Trans. Electron.* **2021**, *104*, 643–650. [CrossRef]
44. Adeniyi, A.E.; Jimoh, R.G.; Awotunde, J.B. A systematic review on elliptic curve cryptography algorithm for internet of things: Categorization, application areas, and security. *Comput. Electr. Eng.* **2024**, *118*, 109330. [CrossRef]
45. Shaaban, M.A.; Alsharkawy, A.S.; AbouKreisha, M.T.; Razek, M.A. Efficient ECC-based authentication scheme for fog-based IoT environment. *arXiv* **2024**, arXiv:2408.02826. [CrossRef]
46. Sudhakaran, P. Energy efficient distributed lightweight authentication and encryption technique for IoT security. *Int. J. Commun. Syst.* **2022**, *35*, e4198. [CrossRef]
47. Sultan, I.; Mir, B.J.; Banday, M.T. Analysis and optimization of advanced encryption standard for the internet of things. In Proceedings of the 2020 7th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 27–28 February 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 571–575.
48. Salman, R.S.; Farhan, A.K.; Shakir, A. Lightweight modifications in the Advanced Encryption Standard (AES) for IoT applications: A comparative survey. In Proceedings of the 2022 International Conference on Computer Science and Software Engineering (CSASE), Duhok, Iraq, 15–17 March 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 325–330.
49. Tanksale, V. Efficient Elliptic Curve Diffie–Hellman Key Exchange for Resource-Constrained IoT Devices. *Electronics* **2024**, *13*, 3631. [CrossRef]
50. Jerbi, W.; Guermazi, A.; Trabelsi, H. Crypto-ECC: a rapid secure protocol for large-scale wireless sensor networks deployed in internet of things. In Proceedings of the Theory and Applications of Dependable Computer Systems: Proceedings of the Fifteenth International Conference on Dependability of Computer Systems DepCoS-RELCOMEX, Brunów, Poland, 29 June–3 July 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 293–303.
51. Nashwan, S. Secure authentication scheme using Diffie–Hellman key agreement for smart IoT irrigation systems. *Electronics* **2022**, *11*, 188. [CrossRef]
52. Ali, S.; Humaria, A.; Ramzan, M.S.; Khan, I.; Saqlain, S.M.; Ghani, A.; Zakia, J.; Alzahrani, B.A. An efficient cryptographic technique using modified Diffie–Hellman in wireless sensor networks. *Int. J. Distrib. Sens. Netw.* **2020**, *16*, 1550147720925772. [CrossRef]
53. Muth, R.; Tschorsch, F. Smartdhx: Diffie–Hellman key exchange with smart contracts. In Proceedings of the 2020 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS), Oxford, UK, 3–6 August 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 164–168.
54. Shamsoshoara, A.; Korenda, A.; Afghah, F.; Zeadally, S. A survey on physical unclonable function (PUF)-based security solutions for Internet of Things. *Comput. Netw.* **2020**, *183*, 107593. [CrossRef]
55. Yadav, A.; Kumar, S.; Singh, J. A review of physical unclonable functions (PUFs) and its applications in IoT environment. In *Ambient Communications and Computer Systems: Proceedings of RACCCS 2021, Ajmer, India, 20–21 August 2021*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 1–13.
56. Duan, S.; Sai, G. Protecting sram puf from bti aging-based cloning attack. In Proceedings of the 2022 35th SBC/SBMicro/IEEE/ACM Symposium on Integrated Circuits and Systems Design (SBCCI), Porto Alegre, Brazil, 22–26 August 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6.
57. Lounis, K.; Zulkernine, M. Lessons learned: Analysis of PUF-based authentication protocols for IoT. *Digit. Threat. Res. Pract.* **2023**, *4*, 1–33. [CrossRef]
58. Mostafa, A.; Lee, S.J.; Peker, Y.K. Physical unclonable function and hashing are all you need to mutually authenticate IoT devices. *Sensors* **2020**, *20*, 4361. [CrossRef] [PubMed]
59. Alkatheiri, M.S.; Sangi, A.R.; Anamalamudi, S. Physical unclonable function (PUF)-based security in Internet of Things (IoT): Key challenges and solutions. In *Handbook of Computer Networks and Cyber Security: Principles and Paradigms*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 461–473.
60. Ning, H.; Farha, F.; Ullah, A.; Mao, L. Physical unclonable function: Architectures, applications and challenges for dependable security. *IET Circuits Devices Syst.* **2020**, *14*, 407–424. [CrossRef]
61. Gilani, K.; Bertin, E.; Hatin, J.; Crespi, N. A survey on blockchain-based identity management and decentralized privacy for personal data. In Proceedings of the 2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), Paris, France, 28–30 September 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 97–101.
62. Alanzi, H.; Alkhatib, M. Towards improving privacy and security of identity management systems using blockchain technology: A systematic review. *Appl. Sci.* **2022**, *12*, 12415. [CrossRef]
63. Luecking, M.; Fries, C.; Lamberti, R.; Stork, W. Decentralized identity and trust management framework for Internet of Things. In Proceedings of the 2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Toronto, ON, Canada, 2–6 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–9.

64. Kemmoe, V.Y.; Stone, W.; Kim, J.; Kim, D.; Son, J. Recent advances in smart contracts: A technical overview and state of the art. *IEEE Access* **2020**, *8*, 117782–117801. [CrossRef]
65. Maitra, S.; Yanambaka, V.P.; Puthal, D.; Abdelgawad, A.; Yelamarthi, K. Integration of Internet of Things and blockchain toward portability and low-energy consumption. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e4103. [CrossRef]
66. Gonçalves, C.; Sousa, B.; Vukovic, M.; Kusek, M. A federated authentication and authorization approach for IoT farming. *Internet Things* **2023**, *22*, 100785. [CrossRef]
67. Mahalle, P.N.; Railkar, P.N. *Identity Management for Internet of Things*; River Publishers: Aalborg, Denmark, 2022.
68. Liu, Y.; He, D.; Obaidat, M.S.; Kumar, N.; Khan, M.K.; Choo, K.K.R. Blockchain-based identity management systems: A review. *J. Netw. Comput. Appl.* **2020**, *166*, 102731. [CrossRef]
69. Pöhn, D.; Hillmann, P. Reference service model for federated identity management. In Proceedings of the International Conference on Business Process Modeling, Development and Support, Melbourne, VIC, Australia, 28–29 June 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 196–211.
70. Gangwani, P.; Joshi, S.; Upadhyay, H.; Lagos, L. IoT device identity management and blockchain for security and data integrity. *Int. J. Comput. Appl* **2023**, *184*, 49–55. [CrossRef]
71. Adja, Y.C.E.; Hammi, B.; Serhrouchni, A.; Zeadally, S. A blockchain-based certificate revocation management and status verification system. *Comput. Secur.* **2021**, *104*, 102209. [CrossRef]
72. Khodaei, M.; Papadimitratos, P. Scalable & resilient vehicle-centric certificate revocation list distribution in vehicular communication systems. *IEEE Trans. Mob. Comput.* **2020**, *20*, 2473–2489.
73. Huang, H.S.; Jiang, Z.Y.; Cheng, H.T.; Sun, H.M. Hybrid Online Certificate Status Protocol with Certificate Revocation List for Smart Grid Public Key Infrastructure. *arXiv* **2024**, arXiv:2401.10787.
74. Mahmmod, K.F.; Azeez, M.M.; Ismael, Z.H. Design an active verification mechanism for certificates revocation in OCSP for internet authentication. *Int. J. Electr. Comput. Eng.* **2020**, *10*, 4208. [CrossRef]
75. Ogonji, M.M.; Okeyo, G.; Wafula, J.M. A survey on privacy and security of Internet of Things. *Comput. Sci. Rev.* **2020**, *38*, 100312. [CrossRef]
76. Sharma, A.; Pilli, E.S.; Mazumdar, A.P.; Gera, P. Towards trustworthy Internet of Things: A survey on Trust Management applications and schemes. *Comput. Commun.* **2020**, *160*, 475–493. [CrossRef]
77. Rachit; Bhatt, S.; Ragiri, P.R. Security trends in Internet of Things: A survey. *SN Appl. Sci.* **2021**, *3*, 121. [CrossRef]
78. Dritsas, E.; Trigka, M.; Mylonas, P. A Survey on Privacy-Enhancing Techniques in the Era of Artificial Intelligence. In Proceedings of the Novel & Intelligent Digital Systems Conferences, Athens, Greece, 25–27 September 2024; Springer: Berlin/Heidelberg, Germany, 2024; pp. 385–392.
79. Abosata, N.; Al-Rubaye, S.; Inalhan, G.; Emmanouilidis, C. Internet of things for system integrity: A comprehensive survey on security, attacks and countermeasures for industrial applications. *Sensors* **2021**, *21*, 3654. [CrossRef] [PubMed]
80. Anwar, M.R.; Apriani, D.; Adianita, I.R. Hash Algorithm In Verification Of Certificate Data Integrity And Security. *Aptisi Trans. Technopreneurship (ATT)* **2021**, *3*, 181–188. [CrossRef]
81. Fomichev, V.; Bobrovskiy, D.; Koreneva, A.; Nabiev, T.; Zadorozhny, D. Data integrity algorithm based on additive generators and hash function. *J. Comput. Virol. Hacking Tech.* **2022**, *18*, 31–41. [CrossRef]
82. Windarta, S.; Suryadi, S.; Ramli, K.; Pranggono, B.; Gunawan, T.S. Lightweight cryptographic hash functions: Design trends, comparative study, and future directions. *IEEE Access* **2022**, *10*, 82272–82294. [CrossRef]
83. Kavin, B.P.; Ganapathy, S. A new digital signature algorithm for ensuring the data integrity in cloud using elliptic curves. *Int. Arab J. Inf. Technol.* **2021**, *18*, 180–190.
84. Aggarwal, S.; Kumar, N. Digital signatures. In *Advances in Computers*; Elsevier: Amsterdam, The Netherlands, 2021; Volume 121, pp. 95–107.
85. Lalem, F.; Laouid, A.; Kara, M.; Al-Khalidi, M.; Eleyan, A. A novel digital signature scheme for advanced asymmetric encryption techniques. *Appl. Sci.* **2023**, *13*, 5172. [CrossRef]
86. Megouache, L.; Zitouni, A.; Djoudi, M. Ensuring user authentication and data integrity in multi-cloud environment. *Hum. Centric Comput. Inf. Sci.* **2020**, *10*, 1–20. [CrossRef]
87. Garagad, V.G.; Iyer, N.C.; Wali, H.G. Data integrity: A security threat for internet of things and cyber-physical systems. In Proceedings of the 2020 International Conference on Computational Performance Evaluation (ComPE), Shillong, India, 2–4 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 244–249.
88. Li, H.; Kumar, V.; Park, J.M.; Yang, Y. Cumulative message authentication codes for resource-constrained IoT networks. *IEEE Internet Things J.* **2021**, *8*, 11847–11859. [CrossRef]
89. Thakor, V.A.; Razzaque, M.A.; Khandaker, M.R. Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities. *IEEE Access* **2021**, *9*, 28177–28193. [CrossRef]

90. Bhagat, V.; Kumar, S.; Gupta, S.K.; Chaube, M.K. Lightweight cryptographic algorithms based on different model architectures: A systematic review and futuristic applications. *Concurr. Comput. Pract. Exp.* **2023**, *35*, e7425. [CrossRef]
91. Sleem, L.; Couturier, R. Speck-R: An ultra light-weight cryptographic scheme for Internet of Things. *Multimed. Tools Appl.* **2021**, *80*, 17067–17102. [CrossRef]
92. Yang, P.; Xiong, N.; Ren, J. Data security and privacy protection for cloud storage: A survey. *IEEE Access* **2020**, *8*, 131723–131740. [CrossRef]
93. Alharbi, A.; Zamzami, H.; Samkri, E. Survey on homomorphic encryption and address of new trend. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*. [CrossRef]
94. Yang, Z.; Hu, S.; Chen, K. FPGA-based hardware accelerator of homomorphic encryption for efficient federated learning. *arXiv* **2020**, arXiv:2007.10560.
95. Sihotang, H.T.; Efendi, S.; Zamzami, E.M.; Mawengkang, H. Design and implementation of Rivest Shamir Adleman's (RSA) cryptography algorithm in text file data security. *J. Phys. Conf. Ser.* **2020**, *1641*, 012042. [CrossRef]
96. Kumar, V.V.; Pabitha, P. Privacy-Preserving Brakerski-Gentry-Vaikuntanathan (BGV) Homomorphic Encryption for IoMT Data Security. In Proceedings of the 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), Chikkaballapur, India, 18–19 April 2024; IEEE: Piscataway, NJ, USA, 2024; Volume 1, pp. 1–6.
97. Marcolla, C.; Sucasas, V.; Manzano, M.; Bassoli, R.; Fitzek, F.H.; Aaraj, N. Survey on fully homomorphic encryption, theory, and applications. *Proc. IEEE* **2022**, *110*, 1572–1609. [CrossRef]
98. Bavdekar, R.; Chopde, E.J.; Bhatia, A.; Tiwari, K.; Daniel, S.J. Post quantum cryptography: Techniques, challenges, standardization, and directions for future research. *arXiv* **2022**, arXiv:2202.02826.
99. Cho, C.H.; Chen, C.Y.; Chen, K.C.; Huang, T.W.; Hsu, M.C.; Cao, N.P.; Zeng, B.; Tan, S.G.; Chang, C.R. Quantum computation: Algorithms and applications. *Chin. J. Phys.* **2021**, *72*, 248–269. [CrossRef]
100. Bandara, H.; Herath, Y.; Weerasundara, T.; Alawatugoda, J. On advances of lattice-based cryptographic schemes and their implementations. *Cryptography* **2022**, *6*, 56. [CrossRef]
101. Balamurugan, C.; Singh, K.; Ganesan, G.; Rajarajan, M. Post-quantum and code-based cryptography—Some prospective research directions. *Cryptography* **2021**, *5*, 38. [CrossRef]
102. Kuang, R.; Barbeau, M. Indistinguishability and non-deterministic encryption of the quantum safe multivariate polynomial public key cryptographic system. In Proceedings of the 2021 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Virtual, 12–17 September 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–5.
103. Welsh, T.; Benkhelifa, E. On resilience in cloud computing: A survey of techniques across the cloud domain. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 59. [CrossRef]
104. Hamdan, S.; Ayyash, M.; Almajali, S. Edge-computing architectures for internet of things applications: A survey. *Sensors* **2020**, *20*, 6441. [CrossRef] [PubMed]
105. Goyal, H.; Saha, S. Multi-party computation in iot for privacy-preservation. In Proceedings of the 2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS), Bologna, Italy, 10–13 July 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1280–1281.
106. Sahinbas, K.; Catak, F.O. Secure multi-party computation-based privacy-preserving data analysis in healthcare IoT systems. In *Interpretable Cognitive Internet of Things for Healthcare*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 57–72.
107. Kurt, A.; Mercan, S.; Shlomovits, O.; Erdin, E.; Akkaya, K. Lngate: Powering iot with next generation lightning micro-payments using threshold cryptography. In Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks, Virtual, 28 June–2 July 2021; pp. 117–128.
108. Tan, L.; Yu, K.; Yang, C.; Bashir, A.K. A blockchain-based Shamir's threshold cryptography for data protection in industrial internet of things of smart city. In Proceedings of the 1st Workshop on Artificial Intelligence and Blockchain Technologies for Smart Cities with 6G, New Orleans, LA, USA, 25–29 October 2021; pp. 13–18.
109. Hsueh, C.W.; Chin, C.T. Toward Trusted IoT by General Proof-of-Work. *Sensors* **2022**, *23*, 15. [CrossRef] [PubMed]
110. Yazdinejad, A.; Srivastava, G.; Parizi, R.M.; Dehghantanha, A.; Karimipour, H.; Karizno, S.R. Slpow: Secure and low latency proof of work protocol for blockchain in green iot networks. In Proceedings of the 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), Antwerp, Belgium, 25–28 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–5.
111. Mišić, J.; Mišić, V.B.; Chang, X. Design of proof-of-stake PBFT algorithm for IoT environments. *IEEE Trans. Veh. Technol.* **2022**, *72*, 2497–2510. [CrossRef]
112. Singhal, D.; Ahuja, L.; Seth, A. POSMETER: Proof-of-stake blockchain for enhanced smart meter data security. *Int. J. Inf. Technol.* **2024**, *16*, 1171–1184. [CrossRef]
113. Li, Y.; Cao, B.; Peng, M.; Zhang, L.; Zhang, L.; Feng, D.; Yu, J. Direct acyclic graph-based ledger for Internet of Things: Performance and security analysis. *IEEE/ACM Trans. Netw.* **2020**, *28*, 1643–1656. [CrossRef]

114. Bhandary, M.; Parmar, M.; Ambawade, D. A blockchain solution based on directed acyclic graph for IoT data security using IoTA tangle. In Proceedings of the 2020 5th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 10–12 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 827–832.
115. Rahaman, M.; Arya, V.; Orozco, S.M.; Pappachan, P. Secure multi-party computation (SMPC) protocols and privacy. In *Innovations in Modern Cryptography*; IGI Global: Hershey, PA, USA, 2024; pp. 190–214.
116. Fu, X.; Xiong, L.; Li, F.; Yang, X.; Xiong, N. Blockchain-Based Efficiently Privacy-Preserving Federated Learning Framework Using Shamir Secret Sharing. *IEEE Trans. Consum. Electron.* **2024**. [CrossRef]
117. Linke, T.; Harth-Kitzerow, C. Optimizations for Secure Multiparty Computation Protocols. *Network* **2022**, *13–16.* [CrossRef]
118. Blackburn, S.R. Combinatorics and threshold cryptography. In *Combinatorial Designs and their Applications*; Routledge: London, UK, 2023; pp. 49–70.
119. Gheisari, M.; Javadpour, A.; Gao, J.; Abbasi, A.A.; Pham, Q.V.; Liu, Y. PPDMIT: A lightweight architecture for privacy-preserving data aggregation in the Internet of Things. *J. Ambient. Intell. Humaniz. Comput.* **2023**, *14*, 5211–5223. [CrossRef]
120. Liu, X.; Wang, X.; Yu, K.; Yang, X.; Ma, W.; Li, G.; Zhao, X. Secure data aggregation aided by privacy preserving in Internet of Things. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 4858722. [CrossRef]
121. Tewari, A.; Gupta, B.B. Security, privacy and trust of different layers in Internet-of-Things (IoTs) framework. *Future Gener. Comput. Syst.* **2020**, *108*, 909–920. [CrossRef]
122. Jiang, B.; Li, J.; Yue, G.; Song, H. Differential privacy for industrial internet of things: Opportunities, applications, and challenges. *IEEE Internet Things J.* **2021**, *8*, 10430–10451. [CrossRef]
123. Husnoo, M.A.; Anwar, A.; Chakrabortty, R.K.; Doss, R.; Ryan, M.J. Differential privacy for IoT-enabled critical infrastructure: A comprehensive survey. *IEEE Access* **2021**, *9*, 153276–153304. [CrossRef]
124. Zheng, Z.; Wang, T.; Bashir, A.K.; Alazab, M.; Mumtaz, S.; Wang, X. A decentralized mechanism based on differential privacy for privacy-preserving computation in smart grid. *IEEE Trans. Comput.* **2021**, *71*, 2915–2926. [CrossRef]
125. Kumar, G.S.; Premalatha, K.; Maheshwari, G.U.; Kanna, P.R.; Vijaya, G.; Nivaashini, M. Differential privacy scheme using Laplace Mechanism and statistical method computation in deep neural network for privacy preservation. *Eng. Appl. Artif. Intell.* **2024**, *128*, 107399. [CrossRef]
126. Dong, J.; Roth, A.; Su, W.J. Gaussian differential privacy. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2022**, *84*, 3–37. [CrossRef]
127. Zhao, Y.; Zhao, J.; Yang, M.; Wang, T.; Wang, N.; Lyu, L.; Niyato, D.; Lam, K.Y. Local differential privacy-based federated learning for internet of things. *IEEE Internet Things J.* **2020**, *8*, 8836–8853. [CrossRef]
128. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Poor, H.V. Federated learning for internet of things: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1622–1658. [CrossRef]
129. Briggs, C.; Fan, Z.; Andras, P. A review of privacy-preserving federated learning for the Internet-of-Things. In *Federated Learning Systems. Studies in Computational Intelligence*; Springer: Cham, Switzerland, 2021; pp. 21–50.
130. Singh, S.; Sharma, P.K.; Moon, S.Y.; Park, J.H. Advanced lightweight encryption algorithms for IoT devices: Survey, challenges and solutions. *J. Ambient. Intell. Humaniz. Comput.* **2024**, *15*, 1625–1642. [CrossRef]
131. Teng, L.; Li, H.; Yin, S.; Sun, Y. A Modified Advanced Encryption Standard for Data Security. *Int. J. Netw. Secur.* **2020**, *22*, 112–117.
132. Muttaqin, K.; Rahmadoni, J. Analysis and design of file security system AES (advanced encryption standard) cryptography based. *J. Appl. Eng. Technol. Sci. (JAETS)* **2020**, *1*, 113–123. [CrossRef]
133. Nayancy; Dutta, S.; Chakraborty, S. A survey on implementation of lightweight block ciphers for resource constraints devices. *J. Discret. Math. Sci. Cryptogr.* **2022**, *25*, 1377–1398. [CrossRef]
134. Raza, A.R.; Mahmood, K.; Amjad, M.F.; Abbas, H.; Afzal, M. On the efficiency of software implementations of lightweight block ciphers from the perspective of programming languages. *Future Gener. Comput. Syst.* **2020**, *104*, 43–59. [CrossRef]
135. Geppert, T.; Deml, S.; Sturzenegger, D.; Ebert, N. Trusted execution environments: Applications and organizational challenges. *Front. Comput. Sci.* **2022**, *4*, 930741. [CrossRef]
136. Fan, Y.; Liu, S.; Tan, G.; Qiao, F. Fine-grained access control based on trusted execution environment. *Future Gener. Comput. Syst.* **2020**, *109*, 551–561. [CrossRef]
137. Zhang, Y.; Wang, Z.; Cao, J.; Hou, R.; Meng, D. ShuffleFL: Gradient-preserving federated learning using trusted execution environment. In Proceedings of the 18th ACM international Conference on Computing Frontiers, Virtual, 11–13 May 2021; pp. 161–168.
138. Gao, Y.; Al-Sarawi, S.F.; Abbott, D. Physical unclonable functions. *Nat. Electron.* **2020**, *3*, 81–91. [CrossRef]
139. Al-Meer, A.; Al-Kuwari, S. Physical unclonable functions (PUF) for IoT devices. *ACM Comput. Surv.* **2023**, *55*, 314. [CrossRef]
140. Cambou, B.; Philabaum, C.; Booher, D.; Telesca, D.A. Response-based cryptographic methods with ternary physical unclonable functions. In *Advances in Information and Communication: Proceedings of the 2019 Future of Information and Communication Conference (FICC), San Francisco, CA, USA, 14–15 March 2019*; Springer: Berlin/Heidelberg, Germany, 2020; Volume 2, pp. 781–800.

141. Gong, X.; Chen, Y.; Yang, W.; Mei, G.; Wang, Q. InverseNet: Augmenting Model Extraction Attacks with Training Data Inversion. In Proceedings of the IJCAI, Montreal, QC, Canada, 19–27 August 2021; pp. 2439–2447.
142. Zhang, Y.; Jia, R.; Pei, H.; Wang, W.; Li, B.; Song, D. The secret revealer: Generative model-inversion attacks against deep neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 253–261.
143. Hong, S.; Chandrasekaran, V.; Kaya, Y.; Dumitraş, T.; Papernot, N. On the effectiveness of mitigating data poisoning attacks with gradient shaping. *arXiv* **2020**, arXiv:2002.11497.
144. Atlam, H.F.; Wills, G.B. IoT security, privacy, safety and ethics. In *Digital Twin Technologies and Smart Cities*; Springer: Berlin/Heidelberg, Germany, 2020; pp.123–149.
145. Tawalbeh, L.; Muheidat, F.; Tawalbeh, M.; Quwaider, M. IoT Privacy and security: Challenges and solutions. *Appl. Sci.* **2020**, *10*, 4102. [CrossRef]
146. Bayılmış, C.; Ebleme, M.A.; Çavuşoğlu, Ü.; Küçük, K.; Sevin, A. A survey on communication protocols and performance evaluations for Internet of Things. *Digit. Commun. Netw.* **2022**, *8*, 1094–1104. [CrossRef]
147. Pai, G.N.; Pai, M.S.; Gowd, V.D.; Shruthi, M.; Naveen K, B. Internet of Things: A survey on devices, ecosystem, components and communication protocols. In Proceedings of the 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 5–7 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 611–616.
148. Gerodimos, A.; Maglaras, L.; Ferrag, M.A.; Ayres, N.; Kantzavelou, I. IoT: Communication protocols and security threats. *Internet Things Cyber Phys. Syst.* **2023**, *3*, 1–13. [CrossRef]
149. Yalçınkaya, F.; Aydilek, H.; Erten, M.Y.; İnanç, N. IoT based smart home testbed using MQTT communication protocol. *Int. J. Eng. Res. Dev.* **2020**, *12*, 317–324. [CrossRef]
150. Shanmugapriya, D.; Patel, A.; Srivastava, G.; Lin, J.C.W. MQTT protocol use cases in the Internet of Things. In Proceedings of the Big Data Analytics: 9th International Conference, BDA 2021, Virtual, 15–18 December 2021; Proceedings 9; Springer: Berlin/Heidelberg, Germany, 2021; pp. 146–162.
151. Chen, F.; Huo, Y.; Zhu, J.; Fan, D. A review on the study on MQTT security challenge. In Proceedings of the 2020 IEEE International Conference on Smart Cloud (SmartCloud), Washington, DC, USA, 6–8 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 128–133.
152. Alhaidari, F.A.; Alqahtani, E.J. Securing Communication between Fog Computing and IoT Using Constrained Application Protocol (CoAP): A Survey. *J. Commun.* **2020**, *15*, 14–30. [CrossRef]
153. Bansal, S.; Kumar, D. A Reliable CoAP Protocol for IoT Communication. 2022. Available online: https://www.researchsquare.com/article/rs-1974849/v1 (accessed on 8 January 2025).
154. Azeez, H.H.; Abdullah, M.Z. Performance analysis of constrained application protocol (CoAP). *AIP Conf. Proc.* **2023**, *2591*, 030074.
155. Pereira, D.S.; De Morais, M.R.; Nascimento, L.B.; Alsina, P.J.; Santos, V.G.; Fernandes, D.H.; Silva, M.R. Zigbee protocol-based communication network for multi-unmanned aerial vehicle networks. *IEEE Access* **2020**, *8*, 57762–57771. [CrossRef]
156. Zohourian, A.; Dadkhah, S.; Neto, E.C.P.; Mahdikhani, H.; Danso, P.K.; Molyneaux, H.; Ghorbani, A.A. IoT Zigbee device security: A comprehensive review. *Internet Things* **2023**, *22*, 100791. [CrossRef]
157. Ding, S.; Liu, J.; Yue, M. The use of ZigBee wireless communication technology in industrial automation control. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 8317862. [CrossRef]
158. Ghori, M.R.; Wan, T.C.; Sodhy, G.C. Bluetooth low energy mesh networks: Survey of communication and security protocols. *Sensors* **2020**, *20*, 3590. [CrossRef] [PubMed]
159. Lilli, M.; Braghin, C.; Riccobene, E. Formal Proof of a Vulnerability in Z-Wave IoT Protocol. In Proceedings of the SECRYPT, Virtual, 6–8 July 2021; pp. 198–209.
160. Banti, K.; Karampelia, I.; Dimakis, T.; Boulogeorgos, A.A.A.; Kyriakidis, T.; Louta, M. LoRaWAN communication protocols: A comprehensive survey under an energy efficiency perspective. *Telecom* **2022**, *3*, 322–357. [CrossRef]
161. Dahyan, K.O.; Khattak, S.B.A.; Nasralla, M.M.; Esmail, M.A.; Iqbal, M. Enabling Smart Sensing Systems with Thread Protocol for IoT Connectivity and Cloud Integration. In Proceedings of the International Conference on Sustainability: Developments and Innovations, Riyadh, Saudi Arabia, 18–22 February 2024; Springer: Berlin/Heidelberg, Germany, 2024; pp. 127–134.
162. Tarish, H.A. Enhanced IoT Wi-Fi protocol standard's security using secure remote password. *Period. Eng. Nat. Sci. (PEN)* **2022**, *10*, 632–644. [CrossRef]
163. Wong, H.C. *Man-in-the-Middle Attacks on MQTT Based IoT Networks*; Missouri University of Science and Technology: Rolla, MO, USA, 2022.
164. Vaccari, I.; Aiello, M.; Cambiaso, E. SlowITe, a novel denial of service attack affecting MQTT. *Sensors* **2020**, *20*, 2932. [CrossRef] [PubMed]

165. Boppana, T.K.; Bagade, P. Security risks in MQTT-Based industrial IoT applications. In Proceedings of the 2022 IEEE International Conference on Omni-layer Intelligent Systems (COINS), Barcelona, Spain, 1–3 August 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–5.
166. Shilpa, V.; Vidya, A.; Pattar, S. MQTT based secure transport layer communication for mutual authentication in IoT network. *Glob. Transitions Proc.* **2022**, *3*, 60–66.
167. Tsai, W.C.; Tsai, T.H.; Wang, T.J.; Chiang, M.L. Automatic key update mechanism for lightweight M2M communication and enhancement of iot security: A case study of CoAP using libcoap library. *Sensors* **2022**, *22*, 340. [CrossRef] [PubMed]
168. Sara, A.; Randa, J. Data protection in IoT using CoAP based on enhanced DTLS. *AIP Conf. Proc.* **2024**, *2729*, 040003.
169. Timiraos, M.; Michelena, Á.; Díaz-Longueira, A.; Jove, E.; Aveleira-Mata, J.; García-Rodiguez, I.; Bayón-Gutiérrez, M.; Alaiz-MoretÓn, H.; Calvo-Rolle, J.L. Categorization of CoAP DoS Attack Based on One-Class Boundary Methods. In Proceedings of the International Conference on Soft Computing Models in Industrial and Environmental Applications, Salamanca, Spain, 5–7 September 2023; Springer: Berlin/Heidelberg, Germany, 2023; pp. 112–121.
170. Mishra, V.M.; Kumar, A.; Ompal. FPGA integrated IEEE 802.15. 4 ZigBee wireless sensor nodes performance for industrial plant monitoring and automation. *Nucl. Eng. Technol.* **2022**, *54*, 2444–2452.
171. Farha, F.; Ning, H.; Yang, S.; Xu, J.; Zhang, W.; Choo, K.-K.R. Timestamp scheme to mitigate replay attacks in secure ZigBee networks. *IEEE Trans. Mob. Comput.* **2020**, *21*, 342–351. [CrossRef]
172. Coboi, A.E.; Nguyen, V.; Nguyen, M.; Duy, N.; Tran, T. An Analysis of ZigBee Technologies for Data Routing in Wireless Sensor Networks. *ICSES Trans. Comput. Netw. Commun. (ITCNC)* **2021**.
173. Lacava, A.; Zottola, V.; Bonaldo, A.; Cuomo, F.; Basagni, S. Securing Bluetooth Low Energy networking: An overview of security procedures and threats. *Comput. Netw.* **2022**, *211*, 108953. [CrossRef]
174. Madugula, S.S.; Wei, R. Improvement of Passkey Entry Protocol for Secure Simple Pairing. In Proceedings of the 2023 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Jiangsu, China, 2–4 November 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 347–356.
175. Braghin, C.; Lilli, M.; Riccobene, E. A model-based approach for vulnerability analysis of IoT security protocols: The Z-Wave case study. *Comput. Secur.* **2023**, *127*, 103037. [CrossRef]
176. Nkuba, C.K.; Woo, S.; Lee, H.; Dietrich, S. ZMAD: Lightweight Model-Based Anomaly Detection for the Structured Z-Wave Protocol. *IEEE Access* **2023**, *11*, 60562–60577. [CrossRef]
177. Hessel, F.; Almon, L.; Hollick, M. LoRaWAN security: an evolvable survey on vulnerabilities, attacks and their systematic mitigation. *ACM Trans. Sens. Netw.* **2023**, *18*, 70. [CrossRef]
178. Gaffurini, M.; Flammini, A.; Ferrari, P.; Fernandes Carvalho, D.; Godoy, E.P.; Sisinni, E. End-to-End Emulation of LoRaWAN Architecture and Infrastructure in Complex Smart City Scenarios Exploiting Containers. *Sensors* **2024**, *24*, 2024. [CrossRef]
179. Akestoridis, D.G.; Sekar, V.; Tague, P. On the security of thread networks: Experimentation with openthread-enabled devices. In Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks, San Antonio, TX, USA, 16–19 May 2022; pp. 233–244.
180. Darroudi, S.M.; Gomez, C. Experimental evaluation of 6blemesh: Ipv6-based ble mesh networks. *Sensors* **2020**, *20*, 4623. [CrossRef] [PubMed]
181. Baray, E.; Ojha, N.K. WLAN security protocols and WPA3 security approach measurement through aircrack-ng technique. In Proceedings of the 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 8–10 April 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 23–30.
182. Kwon, S.; Choi, H.K. Evolution of Wi-Fi protected access: security challenges. *IEEE Consum. Electron. Mag.* **2020**, *10*, 74–81. [CrossRef]
183. Akshatha, P.; Dilip Kumar, S. Enhancing security mechanism of MQTT protocol using payload encryption. In Proceedings of the International Conference on Frontiers in Computing and Systems, Bara Phool, India, 19–21 December 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 199–208.
184. Michaelides, M.; Sengul, C.; Patras, P. An experimental evaluation of MQTT authentication and authorization in IoT. In Proceedings of the 15th ACM Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization, New Orleans, LA, USA, 31 January–4 February 2022; pp. 69–76.
185. Sun, L.; Zhu, H. The Research on Security Technology of Earthquake Warning Information Release Based on Zero Trust MQTT Protocol. In Proceedings of the 2023 International Conference on Mobile Internet, Cloud Computing and Information Security (MICCIS), Nanjing, China, 7–9 April 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 131–135.
186. Shah, V. Exploit DTLS Vulnerabilities & Provide a Novel approach to Protect DTLS in CoAP based IoT. *Int. J. Res. Appl. Sci. Eng. Technol.* **2020**, *8*, 216–221.

187. Gunnarsson, M.; Brorsson, J.; Palombini, F.; Seitz, L.; Tiloca, M. Evaluating the performance of the OSCORE security protocol in constrained IoT environments. *Internet Things* **2021**, *13*, 100333. [CrossRef]
188. Höglund, R.; Tiloca, M.; Bouget, S.; Raza, S. Key Update for the IoT Security Standard OSCORE. In Proceedings of the 2023 IEEE International Conference on Cyber Security and Resilience (CSR), Venice, Italy, 31 July–2 August 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 78–85.
189. Nguyen, C.V.; Coboi, A.E.; Bach, N.V.; Dang, A.; Le, T.; Nguyen, H.P.; Nguyen, M.T. ZigBee based data collection in wireless sensor networks. *Int. J. Inf. Commun. Technol.* **2021**, *10*, 211–224. [CrossRef]
190. Muñoz-Calderón, M.; Moh, M. Quantum-Resistant Authentication for Smart Grid: The Case for Using Merkle Trees. In *Handbook of Research on Technical, Privacy, and Security Challenges in a Modern World*; IGI Global: Hershey, PA, USA, 2022; pp. 371–395.
191. Mattsson, J.P.; Smeets, B.; Thormarker, E. Quantum-resistant cryptography. *arXiv* **2021**, arXiv:2112.00399.
192. Ja'afreh, M.A.; Adhami, H.; Alchalabi, A.E.; Hoda, M.; El Saddik, A. Toward integrating software defined networks with the Internet of Things: A review. *Clust. Comput.* **2022**, *25*, 1619–1636. [CrossRef] [PubMed]
193. Mohammed, A.H.; Khaleefah, R.M.; Abdulateef, I.A.; Hussein, M.k. A review software defined networking for internet of things. In Proceedings of the 2020 International Congress on Human–Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 26–28 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–8.
194. Al Hayajneh, A.; Bhuiyan, M.Z.A.; McAndrew, I. Improving internet of things (IoT) security with software-defined networking (SDN). *Computers* **2020**, *9*, 8. [CrossRef]
195. Bekri, W.; Jmal, R.; Chaari Fourati, L. Internet of things management based on software defined networking: a survey. *Int. J. Wirel. Inf. Netw.* **2020**, *27*, 385–410. [CrossRef]
196. Spathoulas, G.; Negka, L.; Pandey, P.; Katsikas, S. Can Blockchain Technology Enhance Security and Privacy in the Internet of Things? In *Advances in Core Computer Science-Based Technologies: Papers in Honor of Professor Nikolaos Alexandris*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 199–228.
197. Alshaikhli, M.; Elfouly, T.; Elharrouss, O.; Mohamed, A.; Ottakath, N. Evolution of Internet of Things from blockchain to IOTA: A survey. *IEEE Access* **2021**, *10*, 844–866. [CrossRef]
198. Hellani, H.; Sliman, L.; Samhat, A.E.; Exposito, E. Tangle the blockchain: Towards connecting blockchain and DAG. In Proceedings of the 2021 IEEE 30th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Bayonne, France, 27–29 October 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 63–68.
199. Tournier, J.; Lesueur, F.; Mouël, F.L.; Guyon, L.; Ben-Hassine, H. IoTMap: A protocol-agnostic multi-layer system to detect application patterns in IoT networks. In Proceedings of the 10th International Conference on the Internet of Things, Malmö, Sweden, 5–9 October 2020; pp. 1–8.
200. Morales, G.A.; Bienek-Parrish, A.; Jenkins, P.; Slavin, R. Protocol-agnostic IoT Device Classification on Encrypted Traffic Using Link-Level Flows. In Proceedings of the Cyber-Physical Systems and Internet of Things Week 2023, San Antonio, TX, USA, 9–12 May 2023; pp. 19–24.
201. Sahu, S.K.; Mohapatra, D.P.; Barik, D.R. An Exhaustive Survey of Privacy and Security Based on IoT Networks. In *IoT Applications, Security Threats, and Countermeasures*; CRC Press: Boca Raton, FL, USA, 2021; pp. 209–226.
202. Mohanta, B.K.; Jena, D.; Satapathy, U.; Patnaik, S. Survey on IoT security: Challenges and solution using machine learning, artificial intelligence and blockchain technology. *Internet Things* **2020**, *11*, 100227. [CrossRef]
203. Sarker, I.H.; Khan, A.I.; Abushark, Y.B.; Alsolami, F. Internet of things (iot) security intelligence: a comprehensive overview, machine learning solutions and research directions. *Mob. Netw. Appl.* **2023**, *28*, 296–312. [CrossRef]
204. Wu, H.; Han, H.; Wang, X.; Sun, S. Research on artificial intelligence enhancing internet of things security: A survey. *IEEE Access* **2020**, *8*, 153826–153848. [CrossRef]
205. DeMedeiros, K.; Hendawi, A.; Alvarez, M. A survey of AI-based anomaly detection in IoT and sensor networks. *Sensors* **2023**, *23*, 1352. [CrossRef] [PubMed]
206. Díaz-Verdejo, J.; Muñoz-Calle, J.; Estepa Alonso, A.; Estepa Alonso, R.; Madinabeitia, G. On the detection capabilities of signature-based intrusion detection systems in the context of web attacks. *Appl. Sci.* **2022**, *12*, 852. [CrossRef]
207. Adekunle, T.S.; Alabi, O.O.; Lawrence, M.O.; Adeleke, T.A.; Afolabi, O.S.; Ebong, G.N.; Egbedokun, G.O.; Bamisaye, T.A. An intrusion system for internet of things security breaches using machine learning techniques. In Proceedings of the Artificial Intelligence and Applications, Corfu, Greece, 27–30 June 2024, Vol. 2; pp. 188–194.
208. Ibor, A.E.; Oladeji, F.A.; Okunoye, O.B.; Ekabua, O.O. An Improved Cyberattack Prediction Technique With Intelligent Clustering And Deep Neural Network. *FUW Trends Sci. Technol. J.* **2020**, *5*, 15–22.
209. Flávio, M.; do Prado, C.B.; da Costa Carmo, L.F.R.; de Sá, A.O.; Ferrari, P.; Pasetti, M. Autoencoder-based Approach to Detect Stealth Cyberattacks in Battery Energy Storage Systems. In Proceedings of the 2024 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4. 0 & IoT), Florence, Italy, 29–31 May 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 452–457.

210. Peppes, N.; Alexakis, T.; Demestichas, K.; Adamopoulou, E. A comparison study of generative adversarial network architectures for malicious cyber-attack data generation. *Appl. Sci.* **2023**, *13*, 7106. [CrossRef]
211. Abdallah, E.E.; Otoom, A.F.; Eleisah, W. Intrusion detection systems using supervised machine learning techniques: a survey. *Procedia Comput. Sci.* **2022**, *201*, 205–212. [CrossRef]
212. Baniasadi, S.; Rostami, O.; Martín, D.; Kaveh, M. A novel deep supervised learning-based approach for intrusion detection in IoT systems. *Sensors* **2022**, *22*, 4459. [CrossRef]
213. Mebawondu, J.O.; Alowolodu, O.D.; Mebawondu, J.O.; Adetunmbi, A.O. Network intrusion detection system using supervised learning paradigm. *Sci. Afr.* **2020**, *9*, e00497. [CrossRef]
214. Alotaibi, Y.; Ilyas, M. Ensemble-learning framework for intrusion detection to enhance internet of things' devices security. *Sensors* **2023**, *23*, 5568. [CrossRef] [PubMed]
215. Thakkar, A.; Lohiya, R. Attack classification of imbalanced intrusion data for IoT network using ensemble-learning-based deep neural network. *IEEE Internet Things J.* **2023**, *10*, 11888–11895. [CrossRef]
216. Alghanam, O.A.; Almobaideen, W.; Saadeh, M.; Adwan, O. An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning. *Expert Syst. Appl.* **2023**, *213*, 118745. [CrossRef]
217. Zaman, S.; Alhazmi, K.; Aseeri, M.A.; Ahmed, M.R.; Khan, R.T.; Kaiser, M.S.; Mahmud, M. Security threats and artificial intelligence based countermeasures for internet of things networks: A comprehensive survey. *IEEE Access* **2021**, *9*, 94668–94690. [CrossRef]
218. Georgescu, T.M. Natural language processing model for automatic analysis of cybersecurity-related documents. *Symmetry* **2020**, *12*, 354. [CrossRef]
219. Zhang, W.E.; Sheng, Q.Z.; Alhazmi, A.; Li, C. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Trans. Intell. Syst. Technol. (TIST)* **2020**, *11*, 24. [CrossRef]
220. Wang, Y.; Wu, H.; Dong, J.; Liu, Y.; Long, M.; Wang, J. Deep Time Series Models: A Comprehensive Survey and Benchmark. *arXiv* **2024**, arXiv:2407.13278.
221. Liu, Z.; Zhu, Z.; Gao, J.; Xu, C. Forecast methods for time series data: a survey. *IEEE Access* **2021**, *9*, 91896–91912. [CrossRef]
222. Rangaraju, S. Secure by intelligence: Enhancing products with AI-driven security measures. *EPH—Int. J. Sci. Eng.* **2023**, *9*, 36–41. [CrossRef]
223. Keshta, I. AI-driven IoT for smart health care: Security and privacy issues. *Inform. Med. Unlocked* **2022**, *30*, 100903. [CrossRef]
224. Dhinakaran, D.; Sankar, S.; Selvaraj, D.; Raja, S.E. Privacy-Preserving Data in IoT-based Cloud Systems: A Comprehensive Survey with AI Integration. *arXiv* **2024**, arXiv:2401.00794.
225. Rehan, H. AI-Driven Cloud Security: The Future of Safeguarding Sensitive Data in the Digital Age. *J. Artif. Intell. Gen. Sci. (JAIGS)* **2024**, *1*, 132–151. [CrossRef]
226. Abed, A.K.; Anupam, A. Review of security issues in Internet of Things and artificial intelligence-driven solutions. *Secur. Priv.* **2023**, *6*, e285. [CrossRef]
227. Mayrhofer, R.; Sigg, S. Adversary models for mobile device authentication. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 198. [CrossRef]
228. Badhib, A.; Alshehri, S.; Cherif, A. A robust device-to-device continuous authentication protocol for the internet of things. *IEEE Access* **2021**, *9*, 124768–124792. [CrossRef]
229. Kumar, D.K.; Reddy, K.K.; Kathrine, G.J.W. Smart Grid Protection with AI and Cryptographic Security. In Proceedings of the 2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 5–7 June 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 246–251.
230. Ishtaiwi, A.; Al Khaldy, M.A.; Al-Qerem, A.; Aldweesh, A.; Almomani, A. Artificial Intelligence in Cryptographic Evolution: Bridging the Future of Security. In *Innovations in Modern Cryptography*; IGI Global: Hershey, PA, USA, 2024; pp. 31–54.
231. Nitaj, A.; Rachidi, T. Applications of neural network-based AI in cryptography. *Cryptography* **2023**, *7*, 39. [CrossRef]
232. Okey, O.D.; Maidin, S.S.; Lopes Rosa, R.; Toor, W.T.; Carrillo Melgarejo, D.; Wuttisittikulkij, L.; Saadi, M.; Zegarra Rodríguez, D. Quantum key distribution protocol selector based on machine learning for next-generation networks. *Sustainability* **2022**, *14*, 15901. [CrossRef]
233. Radanliev, P. Artificial intelligence and quantum cryptography. *J. Anal. Sci. Technol.* **2024**, *15*, 4. [CrossRef]
234. Al-Mohammed, H.A.; Al-Ali, A.; Yaacoub, E.; Qidwai, U.; Abualsaud, K.; Rzewuski, S.; Flizikowski, A. Machine learning techniques for detecting attackers during quantum key distribution in IoT networks with application to railway scenarios. *IEEE Access* **2021**, *9*, 136994–137004. [CrossRef]
235. Lin, J.; Dang, L.; Rahouti, M.; Xiong, K. Ml attack models: adversarial attacks and data poisoning attacks. *arXiv* **2021**, arXiv:2112.02797.

236. Pierazzi, F.; Pendlebury, F.; Cortellazzi, J.; Cavallaro, L. Intriguing properties of adversarial ml attacks in the problem space. In Proceedings of the 2020 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 18–21 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1332–1349.
237. Sadeghi, K.; Banerjee, A.; Gupta, S.K. A system-driven taxonomy of attacks and defenses in adversarial machine learning. *IEEE Trans. Emerg. Top. Comput. Intell.* **2020**, *4*, 450–467. [CrossRef] [PubMed]
238. Jmila, H.; Khedher, M.I. Adversarial machine learning for network intrusion detection: A comparative study. *Comput. Netw.* **2022**, *214*, 109073. [CrossRef]
239. Sarker, I.H. Multi-aspects AI-based modeling and adversarial learning for cybersecurity intelligence and robustness: A comprehensive overview. *Secur. Priv.* **2023**, *6*, e295. [CrossRef]
240. Hathaliya, J.J.; Tanwar, S.; Sharma, P. Adversarial learning techniques for security and privacy preservation: A comprehensive review. *Secur. Priv.* **2022**, *5*, e209. [CrossRef]
241. Zhang, Z.; Al Hamadi, H.; Damiani, E.; Yeun, C.Y.; Taher, F. Explainable artificial intelligence applications in cyber security: State-of-the-art in research. *IEEE Access* **2022**, *10*, 93104–93139. [CrossRef]
242. Chamola, V.; Hassija, V.; Sulthana, A.R.; Ghosh, D.; Dhingra, D.; Sikdar, B. A review of trustworthy and explainable artificial intelligence (xai). *IEEE Access* **2023**, *11*, 78994–79015. [CrossRef]
243. Linardatos, P.; Papastefanopoulos, V.; Kotsiantis, S. Explainable ai: A review of machine learning interpretability methods. *Entropy* **2020**, *23*, 18. [CrossRef] [PubMed]
244. Tiwari, R. Explainable ai (xai) and its applications in building trust and understanding in ai decision making. *Int. J. Sci. Res. Eng. Manag* **2023**, *7*, 1–13. [CrossRef]
245. Spartalis, C.N.; Semertzidis, T.; Daras, P. Balancing XAI with Privacy and Security Considerations. In Proceedings of the European Symposium on Research in Computer Security, The Hague, The Netherlands, 25–29 September 2023; Springer: Berlin/Heidelberg, Germany, 2023; pp. 111–124.
246. Rana, B.; Singh, Y.; Singh, P.K. A systematic survey on internet of things: Energy efficiency and interoperability perspective. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e4166. [CrossRef]
247. Xu, Z.; Liu, W.; Huang, J.; Yang, C.; Lu, J.; Tan, H. Artificial intelligence for securing IoT services in edge computing: a survey. *Secur. Commun. Netw.* **2020**, *2020*, 8872586. [CrossRef]
248. Wang, C.; Yuan, Z.; Zhou, P.; Xu, Z.; Li, R.; Wu, D.O. The security and privacy of mobile edge computing: An artificial intelligence perspective. *IEEE Internet Things J.* **2023**, *10*, 22008–22032. [CrossRef]
249. Hussein, M.; Mohammed, Y.S.; Galal, A.I.; Abd-Elrahman, E.; Zorkany, M. Smart cognitive IoT devices using multi-layer perception neural network on limited microcontroller. *Sensors* **2022**, *22*, 5106. [CrossRef] [PubMed]
250. Zhu, S.; Ota, K.; Dong, M. Green AI for IIoT: Energy efficient intelligent edge computing for industrial internet of things. *IEEE Trans. Green Commun. Netw.* **2021**, *6*, 79–88. [CrossRef]
251. Jurcut, A.; Niculcea, T.; Ranaweera, P.; Le-Khac, N.A. Security considerations for Internet of Things: A survey. *SN Comput. Sci.* **2020**, *1*, 193. [CrossRef]
252. Ye, C.; Cao, W.; Chen, S. Security challenges of blockchain in Internet of things: Systematic literature review. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e4177. [CrossRef]
253. Nižetić, S.; Šolić, P.; Gonzalez-De, D.L.d.I.; Patrono, L. Internet of Things (IoT): Opportunities, issues and challenges towards a smart and sustainable future. *J. Clean. Prod.* **2020**, *274*, 122877. [CrossRef] [PubMed]
254. Mekala, S.H.; Baig, Z.; Anwar, A.; Zeadally, S. Cybersecurity for Industrial IoT (IIoT): Threats, countermeasures, challenges and future directions. *Comput. Commun.* **2023**, *208*, 294–320. [CrossRef]
255. Hora, A.; Kulkarni, P. Wearables and Cybersecurity: Navigating the Threat Landscape. In Proceedings of the 2024 2nd International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 10–12 July 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 563–567.
256. Gómez-Olmos, J. IoT-enabled Edge Computing for Cybersecurity in Autonomous Vehicles-Challenges and Opportunities: Discusses challenges and opportunities in implementing IoT-enabled edge computing for cybersecurity in Avs. *J. Artif. Intell. Res. Appl.* **2023**, *3*, 1–16.
257. Lombardi, M.; Pascale, F.; Santaniello, D. Internet of things: A general overview between architectures, protocols and applications. *Information* **2021**, *12*, 87. [CrossRef]
258. Schiller, E.; Aidoo, A.; Fuhrer, J.; Stahl, J.; Ziörjen, M.; Stiller, B. Landscape of IoT security. *Comput. Sci. Rev.* **2022**, *44*, 100467. [CrossRef]
259. Singh, I.; Lee, S.W. Self-adaptive and secure mechanism for IoT based multimedia services: a survey. *Multimed. Tools Appl.* **2022**, *81*, 26685–26720. [CrossRef]

260. Carrillo-Mondéjar, J.; Turtiainen, H.; Costin, A.; Martínez, J.L.; Suarez-Tangil, G. Hale-iot: Hardening legacy internet of things devices by retrofitting defensive firmware modifications and implants. *IEEE Internet Things J.* **2022**, *10*, 8371–8394. [CrossRef]
261. Hurst, W.; Shone, N. Critical infrastructure security: Cyber-threats, legacy systems and weakening segmentation. In *Management and Engineering of Critical Infrastructures*; Elsevier: Amsterdam, The Netherlands, 2024; pp. 265–286.
262. Lins, F.A.A.; Vieira, M. Security requirements and solutions for iot gateways: A comprehensive study. *IEEE Internet Things J.* **2020**, *8*, 8667–8679. [CrossRef]
263. Albouq, S.S.; Abi Sen, A.A.; Almashf, N.; Yamin, M.; Alshanqiti, A.; Bahbouh, N.M. A survey of interoperability challenges and solutions for dealing with them in IoT environment. *IEEE Access* **2022**, *10*, 36416–36428. [CrossRef]
264. Rasheed, H. Consideration of Cloud-Web-Concepts for Standardization and Interoperability: A Comprehensive Review for Sustainable Enterprise Systems, AI, and IoT Integration. *J. Inf. Technol. Inform.* **2024**, *3*.
265. Sheron, P.F.; Sridhar, K.; Baskar, S.; Shakeel, P.M. A decentralized scalable security framework for end-to-end authentication of future IoT communication. *Trans. Emerg. Telecommun. Technol.* **2020**, *31*, e3815. [CrossRef]
266. Lee, C.; Ahmed, G. Improving IoT privacy, data protection and security concerns. *Int. J. Technol. Innov. Manag. (IJTIM)* **2021**, *1*, 18–33. [CrossRef]
267. Hellani, H.; Sliman, L.; Samhat, A.E.; Exposito, E. On blockchain integration with supply chain: Overview on data transparency. *Logistics* **2021**, *5*, 46. [CrossRef]
268. Le Nguyen, B.; Lydia, E.L.; Elhoseny, M.; Pustokhina, I.; Pustokhin, D.A.; Selim, M.M.; Nguyen, G.N.; Shankar, K. Privacy preserving blockchain technique to achieve secure and reliable sharing of IoT data. *Comput. Mater. Contin.* **2020**, *65*, 87–107. [CrossRef]
269. Singh, S.; Hosen, A.S.; Yoon, B. Blockchain security attacks, challenges, and solutions for the future distributed iot network. *IEEE Access* **2021**, *9*, 13938–13959. [CrossRef]
270. Vishwakarma, M.; Kesswani, N. DIDS: A Deep Neural Network based real-time Intrusion detection system for IoT. *Decis. Anal. J.* **2022**, *5*, 100142. [CrossRef]
271. Panda, S.S.; Mohanta, B.K.; Dey, M.R.; Satapathy, U.; Jena, D. Distributed ledger technology for securing IoT. In Proceedings of the 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020; IEEE: Piscataway, NJ, USA, 2020, pp. 1–6.
272. Fonyi, S. Overview of 5G security and vulnerabilities. *Cyber Def. Rev.* **2020**, *5*, 117–134.
273. Bhat, S.A.; Sofi, I.B.; Chi, C.Y. Edge computing and its convergence with blockchain in 5G and beyond: Security, challenges, and opportunities. *IEEE Access* **2020**, *8*, 205340–205373. [CrossRef]
274. Zhao, Y.; Hu, N.; Zhao, Y.; Zhu, Z. A secure and flexible edge computing scheme for AI-driven industrial IoT. *Clust. Comput.* **2023**, *26*, 283–301. [CrossRef]
275. Sheeja, S.; Francis, E.G. Intrusion detection system and mitigation of threats in IoT networks using AI techniques: A review. *Eng. Appl. Sci. Res.* **2023**, *50*, 633–645.
276. Hadzovic, S.; Mrdovic, S.; Radonjic, M. A path towards an internet of things and artificial intelligence regulatory framework. *IEEE Commun. Mag.* **2023**, *61*, 90–96. [CrossRef]
277. Adestria, E.; Almubaroq, H.Z. Building International Cooperation in Utilizing the Internet of Things (Iot) for Defense: Towards Better Global Security. *Indones. J. Interdiscip. Res. Sci. Technol.* **2024**, *2*, 523–530. [CrossRef]
278. Gupta, R.; Gupta, I.; Singh, A.K.; Saxena, D.; Lee, C.N. An iot-centric data protection method for preserving security and privacy in cloud. *IEEE Syst. J.* **2022**, *17*, 2445–2454. [CrossRef]
279. Dhirani, L.L.; Mukhtiar, N.; Chowdhry, B.S.; Newe, T. Ethical dilemmas and privacy issues in emerging technologies: A review. *Sensors* **2023**, *23*, 1151. [CrossRef]
280. Abdulkareem, N.M. The Economic and Environmental Impact of Sustainable Enterprise Systems: Integrating Cloud, Web Technology, Attacks, AI, IoT, and Security. *J. Inf. Technol. Inform.* **2024**, 3.
281. Gopinath, V.; Rao, K.V.; Rao, S.K. A comprehensive analysis of IoT security towards providing a cost-effective solution: a layered approach. *Int. J. Inf. Technol.* **2023**, *15*, 3813–3826. [CrossRef]
282. Choo, K.K.R.; Gai, K.; Chiaraviglio, L.; Yang, Q. A multidisciplinary approach to Internet of Things (IoT) cybersecurity and risk management. *Comput. Secur.* **2021**, *102*, 102136. [CrossRef]
283. Chiara, P.G. The IoT and the new EU cybersecurity regulatory landscape. *Int. Rev. Law, Comput. Technol.* **2022**, *36*, 118–137. [CrossRef]
284. Zikria, Y.B.; Ali, R.; Afzal, M.K.; Kim, S.W. Next-generation internet of things (iot): Opportunities, challenges, and solutions. *Sensors* **2021**, *21*, 1174. [CrossRef] [PubMed]
285. Alajlan, R.; Alhumam, N.; Frikha, M. Cybersecurity for blockchain-based IoT systems: a review. *Appl. Sci.* **2023**, *13*, 7432. [CrossRef]

286. Ding, S.; Tukker, A.; Ward, H. Opportunities and risks of internet of things (IoT) technologies for circular business models: A literature review. *J. Environ. Manag.* **2023**, *336*, 117662. [CrossRef] [PubMed]
287. Wessels, M.; van den Brink, P.; Verburgh, T.; Cadet, B.; van Ruijven, T. Understanding incentives for cybersecurity investments: Development and application of a typology. *Digit. Bus.* **2021**, *1*, 100014. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Review

SoK: Delegated Security in the Internet of Things

Emiliia Geloczi [1,*], Felix Klement [1], Patrick Struck [2] and Stefan Katzenbeisser [1,*]

[1] Faculty of Computer Science and Mathematics, University of Passau, 94032 Passau, Germany; felix.klement@uni-passau.de
[2] Cryptography and Cyber Security, Department of Computer and Information Science, University of Konstanz, 78464 Konstanz, Germany; patrick.struck@uni-konstanz.de
* Correspondence: emiliia.geloczi@uni-passau.de (E.G.); stefan.katzenbeisser@uni-passau.de (S.K.)

Abstract: The increased use of electronic devices in the Internet of Things (IoT) leads not only to an improved comfort of living but also to an increased risk of attacks. IoT security has thus become an important research field. However, due to limits on performance and bandwidth, IoT devices are often not powerful enough to execute, e.g., costly cryptographic algorithms or protocols. This limitation can be solved through a delegation concept. By delegating certain operations to devices with sufficient resources, it is possible to achieve a high level of security without overloading a device that needs protection. In this paper, we give an overview of current approaches for security delegation in the context of IoT, formalise security notions, discuss the security of existing approaches, and identify further research questions. Furthermore, a mathematical formalisation of the CIA triad (confidentiality, integrity, and availability) is proposed for the predefined application areas, in order to evaluate the different approaches.

Keywords: Internet of Things; delegation; security; access control; authorisation; computation; authentication

1. Introduction

The number of electronic devices in use is steadily increasing [1]. While IoT devices enhance comfort and are attractive for widespread utilisation, they also pose a significant cybersecurity risk [2]. During operation, IoT devices communicate with each other and with the outside world, receiving, transmitting and generating a large amount of data, which can be critical under certain conditions. Critical data in possession of adversaries can cause irreparable damage to individuals or institutions [3]. Beyond unauthorised data access, adversaries can also compromise IoT devices, leading to service disruptions and potential safety risks. For example, in healthcare applications, a compromised medical IoT device, e.g., a connected inhaler, blood sensor, or asthma monitor, could generate false alerts or misleading medical recommendations to patients and doctors, resulting in inappropriate treatment and endangering patient safety [4].

Despite IoT devices providing a wide range of services, their computational and communication resources are constrained, as they are designed to be lightweight, and in many cases battery-powered. As a result, they are often unable to execute complex security mechanisms and protect themselves from cyberattacks [5]. One effective strategy to address this issue and ensure the security of resource-constrained IoT devices is delegation [6,7].

The general idea behind security delegation is that a resource-constrained device, which lacks the computational capacity to perform advanced security measures, outsources them to a more powerful device, called a *delegate*. The *delegate* acts as a trusted entity that

performs security-related functions on behalf of the constrained device, ensuring security and avoiding overloading the limited resources, thereby "encapsulating" the device in a secure environment (see Figure 1).

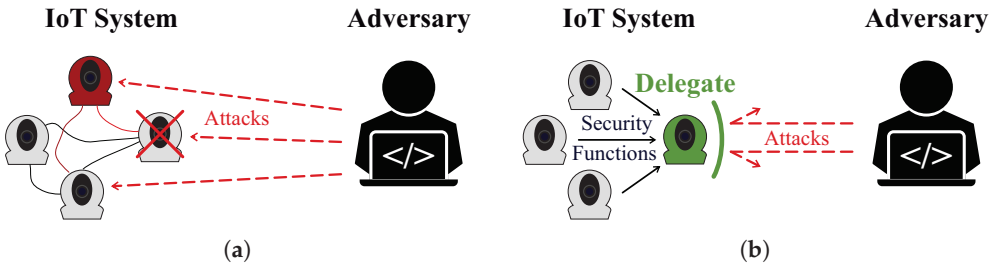


Figure 1. IoT System (**a**) without and (**b**) with security delegation.

For example, Saied et al. propose a collaborative approach to key establishment based on delegation [8]. In the proposed scenario, a highly resource-constrained sensor intends to exchange data with an external server securely. Since the sensor is unable to perform costly computations, it delegates the establishment of a session key with the external server to a group of proxy nodes. The proxies complete key agreement with the server on behalf of the sensor using Diffie–Hellman key exchange and submit the resulting session key to the sensor. Thus, all heavy computations are performed by proxies, while the sensor only needs to handle the lightweight secret key. This approach results in lower costs compared to a TLS handshake. Additionally, one could argue that it shifts the adversary's target from the "weak" device to the "strong" one, which can be better protected. Hence, we can hope that delegation in IoT systems also enhances security. However, delegation may introduce other attack vectors and complicate security models as well as security properties due to the introduction of the delegate.

Several studies have been carried out on security delegation, exploring different techniques and implementations. However, most works use different adversary models and assumptions [9,10]. In some cases, solutions are even proposed without security arguments or proofs [11–13]. Some works focus more on performance improvements [14,15], while others prioritise enhancing security [16,17]. Furthermore, the role of the *delegate* varies, with some approaches relying on centralised authorities [18,19], while others distribute security tasks among multiple entities [8,20]. The listed reasons make a direct comparison of delegating approaches challenging, a difficulty further exacerbated by the lack of a standardised evaluation framework.

1.1. Contributions of This Paper

In this SoK paper, the following key contributions are presented:

- A review of existing works on security delegation in IoT, including the identification of common characteristics and the derivation of generalised architectures for four application domains.
- A flexible adversary model and a formalisation of security properties relevant to the identified application domains.
- An evaluation of the security of approaches described in the literature using the presented security models.
- An identification of the limitations of security delegation in IoT environments and suggestions for future research directions.

1.2. Organisation of This Paper

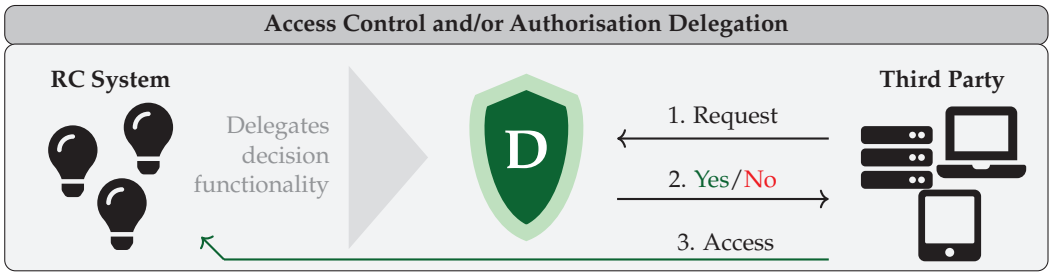
The rest of this paper is organised as follows. Section 2 presents a classification of existing works that introduce delegation-based security solutions in IoT, according

to their application domains. A formal definition of the adversary model is provided in Section 3. Section 4 introduces sound formalisations of security properties for the identified application domains. In Section 5, the existing delegation solutions are evaluated with respect to the proposed security model. The results of the evaluation are discussed in Section 6. Potential future research directions are outlined in Section 7, and the paper is concluded in Section 8.

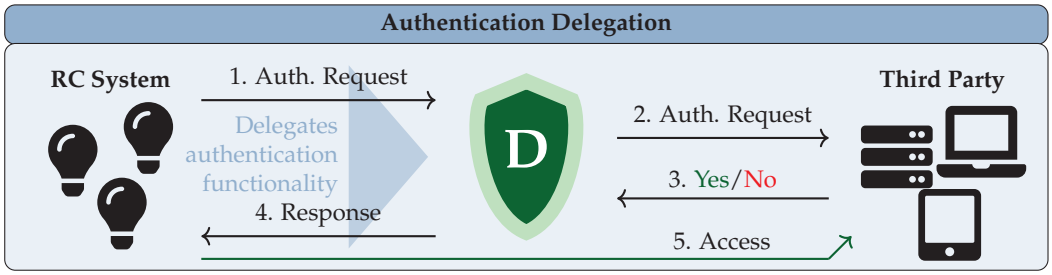
2. Application Domains of Security Delegation

In this paper, we conduct a review of 254 existing research works related to security delegation in the context of the IoT. The methodology used to identify and select these works is described in Appendix A.

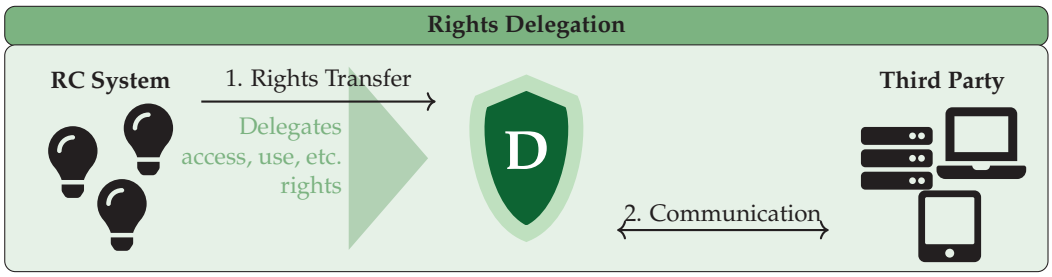
During the analysis of the selected works, it was observed that they focus on four application domains: access control and/or authorisation delegation, authentication delegation, rights delegation, and computation delegation. Within these domains, the proposed solutions exhibit similar architectural patterns, allowing us to identify generalised system architectures presented in Figure 2. These architectures and their illustrative examples are described in detail in this section.



(**a**)

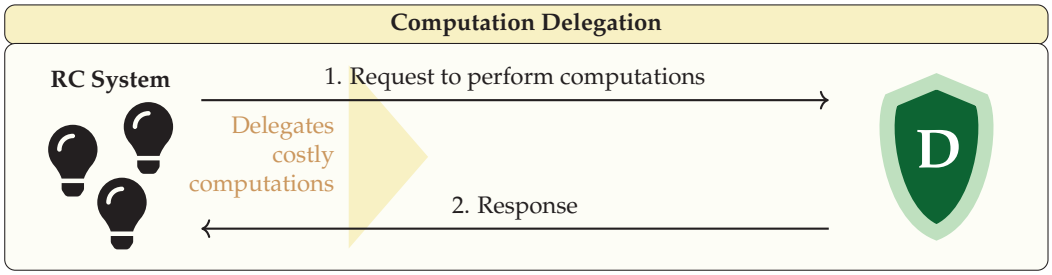


(**b**)



(**c**)

Figure 2. *Cont.*



(**d**)

Figure 2. The generalised schemes of the system architectures based on the application domains: (**a**) Access Control and/or Authorisation Delegation; (**b**) Authentication Delegation; (**c**) Rights Delegation; and (**d**) Computation Delegation.

The identified system architectures include the following parties:

- *Resource-constrained (RC) System:* This party wants to benefit from security delegation. It can be a single device (e.g., a light bulb, a smart vacuum cleaner, a sensor, etc.) or several networked IoT devices. It is assumed that the RC system does not have enough resources to perform some security functions on its own.
- *Delegate:* A device that has sufficient resources to perform resource-intensive cryptographic operations, maintain necessary security and communication protocols, etc. The delegate performs these operations on behalf of the RC system.
- *Third Party:* Any party that interacts with the RC system via the delegate is called a third party. For example, the RC system may want to access resources or services of the third party, or the third party requires access to the RC system.

2.1. Access Control and/or Authorisation Delegation

Access control or authorisation allows or denies access to certain elements of the RC system requested by the third party, based on a security policy. Such policies can be complex decisions that need to be delegated. Through our selection procedure, 47 publications were identified that deal with this domain (see Table A1).

In these applications, the RC system delegates the evaluation of the security policy regarding access control and authorisation to the delegate. The third party first requests access via the delegate, who then verifies the request, makes a decision, and forwards the response to the third party. In case of a positive verdict, the third party obtains access to the RC system (see Figure 2a).

Example

Park and Park present a delegated group-oriented access control security mechanism based on DTLS, which reduces the total computational burden [21]. The proposed system includes the RC system consisting of sensors, a client acting as a third party, and a Security Proxy (SP), which acts as a delegate and manages access control on behalf of the RC system (see Figure 3). During the first step of the proposed protocol, SP forms a list of approved sensor devices and issues a capability ID to each authorised client. Subsequently, the client can directly perform a DTLS handshake and communicate with the desired sensor device from the list. The authors conclude that their mechanism is superior to performing standard approaches without access control delegation in terms of execution time and security.

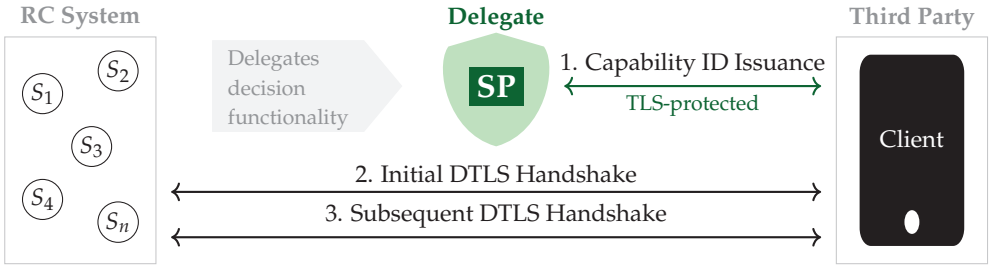


Figure 3. Architecture of the group-oriented end-to-end security mechanism based on DTLS [21]. SP is the Security Proxy that serves as a delegate. S_n is a sensor device.

2.2. Authentication Delegation

Authentication protocols authenticate subjects and distribute secret keys for future communication. A total of 21 papers addressing the delegation of authentication were identified during our literature search (see Table A1).

In this application, the RC system wants to authenticate to a third party. It delegates this process to the delegate who performs authentication on behalf of the RC system and returns the result. Consequently, the RC system obtains credentials to access the third party directly (see Figure 2b).

Example

A delegation-based DTLS framework for Cloud-based IoT Services, called D2TLS, has been proposed by Cho et al. D2TLS allows for establishing secure communication while ensuring secure storage of private keys [22]. Unlike DTLS, D2TLS contains a Security Agent (SA), which does not have access to IoT devices' private keys in order to avoid key escrow problems. The presented solution contains three entities (see Figure 4): a client (RC system), the SA (delegate), and a cloud server (third party). First, the client sends a delegation request to the SA. Then, the SA performs a DTLS handshake with the server on behalf of the client in two steps. Finally, the SA sends the client credentials to access the resources of the server. To evaluate the proposed framework, the authors compare D2TLS with DTLS with respect to delays, energy consumption, code size, memory requirements, and session overhead. In terms of delay and energy consumption, the D2TLS shows significantly better results.

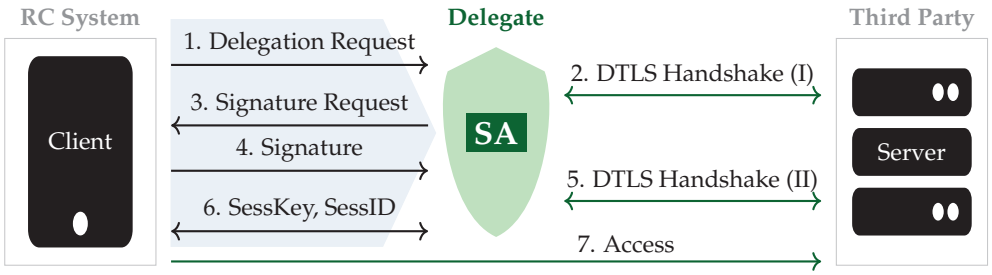


Figure 4. The D2TLS protocol of [22]. SA is a Security Agent.

2.3. Rights Delegation

This category includes papers that address the delegation of various rights (e.g., access, usage, actions, etc.). In total, 123 papers from our literature study targeted this domain (see Table A1).

At first glance, the general architecture of this application domain is similar to authentication delegation. However, there is one significant difference. In authentication delegation, the RC system delegates only the authentication procedure. After its completion, upon receiving the credentials, the RC system independently continues communication with the

third party. In rights delegation, the RC system transfers all its rights related to some third party (e.g., access, use, action, etc.) to the delegate. The delegate is thus able to operate on its own during interaction with the target third party and completely substitute the RC system (see Figure 2c).

Example

Identity-less, asynchronous, and decentralised delegation for IoT based on blockchain technology is presented in [23]. The solution consists of brokers (RC system), buyers (delegate), and a resource manager acting as a third party that offers specific services and resources. All components interact via the blockchain network with smart contracts (see Figure 5). During the preparatory phase, the resource manager registers the offered resources on the blockchain network. The broker has the right to access these resources and can also delegate these rights to the buyer on demand. The buyer requests access rights to the desired resource by triggering a transaction to a delegation smart contract. If the transaction is valid, a new event in the blockchain is generated, and the buyer can request the resource directly. After the first request, the buyer is verified, and only if the verification is successful, the buyer obtains full access to the resource. In comparison with other existing approaches that are not based on blockchain, the described system is asynchronous, decentralised, monotonic, and auditable.

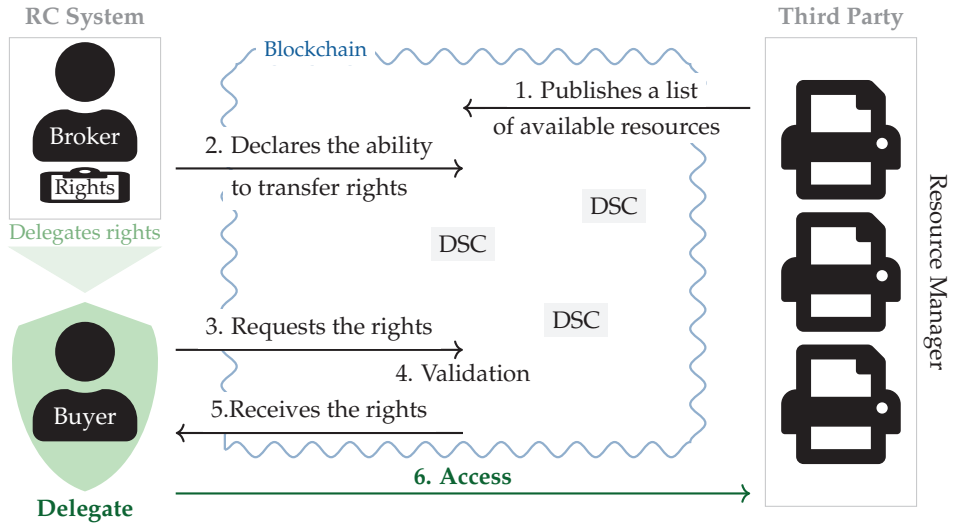


Figure 5. Blockchain-based delegation process of [23]. DSC is a delegation smart contract.

2.4. Computation Delegation

In this application domain, the delegation of computationally intensive cryptographic or computational operations is addressed. Sixty-three papers from our literature review are classified under this domain (see Table A1).

Here, the RC system queries the delegate to perform necessary mathematical computations or cryptographic operations such as key generation, encryption or filtering. After finishing the requested operations, the delegate returns the results to the RC system (see Figure 2d).

Example

In [15], a lightweight revocable hierarchical attribute-based encryption (LW-RHABE) scheme is presented that combines efficient encryption/decryption, flexible key delegation, and revocation.

The solution consists of five main components (see Figure 6). Central and Domain Authorities (CA and DA, respectively) do not participate in delegation directly but generate

system parameters and user secret keys and maintain revocation lists. A Cloud Service Provider (CSP) serves as a delegate that provides storage and computational resources. Data owners and users form the RC system: while data owners are smart IoT devices that collect and encrypt data, users consume data. The workflow of the proposed LW-RHABE system has several steps. First, the CA generates the system's public parameters (a master secret key) and initialises several DAs. Then, the DAs generate secret keys for the users based on their attributes and secret/public keys for data owners. In the next step, data owners create an access control policy, encrypt collected data, and outsource it to the CSP. When the data users want to obtain some data collected by the data owners, they request it and the decryption from the CSP. According to the authors, LW-RHABE allows flexible key delegation and user revocation as well as lightweight encryption and decryption. Based on the results of the performance analysis, the described scheme is significantly more efficient than approaches without delegation.

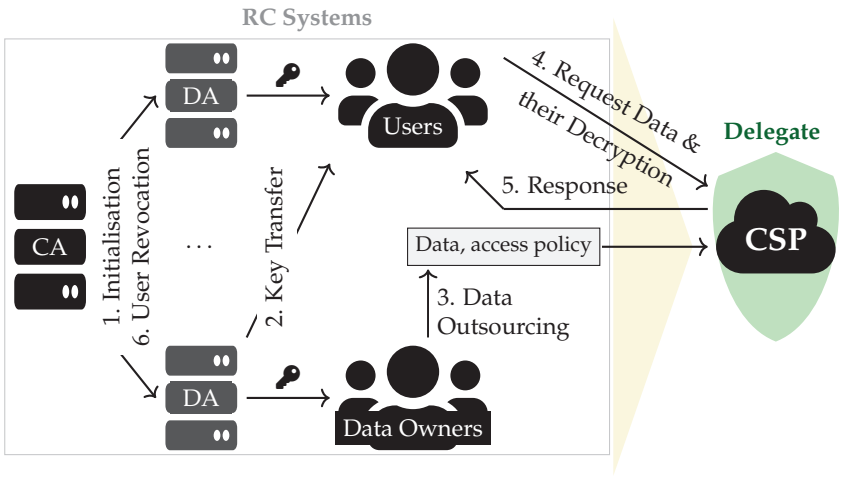


Figure 6. Architecture of the LW-RHABE scheme [15]. CA is Central Authority, DA is Domain Authority, CSP is a Cloud Service Provider.

3. Adversary Model

In order to formulate sound attack models for security delegation in the different application domains mentioned in Section 2, a generalised adversary model is developed, which we believe can provide a possibility to define the different types of adversaries with various capabilities, and thus can be applied even beyond the scope of our study. First, the characteristics and goals of potential adversaries are identified. This leads to a mathematical adversary mode, which is described in detail in this section.

3.1. Adversary Characteristics

Consider three characteristics of an adversary. The first one is the degree of their involvement in the system, where the three following cases can be distinguished: when the adversary is fully external, external with extended capabilities, or internal with direct access to the system. As a second characteristic, the ability of the adversary to corrupt certain parties of the system is taken into account. Here, two cases can be considered: an adversary that either has physical access to the system or does not and thus can or cannot corrupt system parties, respectively. Finally, a third characteristic is related to the adversary's computational resources, represented in terms of the time it takes to complete an attack. All mentioned characteristics are described in Table 1. Combining the cases listed, 18 $(3 \times 3 \times 2)$ adversary types with different capabilities can be defined.

Table 1. Adversary characteristics.

Characteristic	Explanation
Relation to the system ($\mathcal{RS}$)	
External (Ex)	Adversary is located outside the system and has no accomplices who are immediate elements or participants in the system. They are therefore unable to observe processes inside the system.
External+ (Ex^+)	Adversary is located outside the system but has resources or accomplices that allow them, within a limited area, to observe processes inside the system.
Insider (In)	Adversary is a member of the system, so they may have access and knowledge of the protocols used and the processes taking place, as well as awareness of the communication between the objects.
Corruption capabilities ($\mathcal{CC}$)	
No (N)	Adversary has no physical access to the system and no opportunity to corrupt system parties.
Yes (Y)	Adversary has physical access to parts of the system and is able to corrupt desired parties.
Time ($\mathcal{T}$)	
Concrete (C)	Adversary runs in time t and is capable of launching a successful attack with probability at least ϵ against the system.
Asymptotic (As)	Adversary runs in polynomial-time $t(n)$ and is capable of launching a successful attack against the system with probability at least $c + \epsilon(n)$, where ϵ is negligible in the security parameter $n \in \mathbb{N}$ and c is some constant (typically, $c \in \{0, \frac{1}{2}\}$).
Information-theoretic (It)	Adversary has unlimited computation time and is capable of succeeding in an attack against the system with non-negligible probability $\epsilon(n)$ for security parameter $n \in \mathbb{N}$.

3.2. Adversary Goals

There are many possible goals of adversaries; however, in the context of security delegation, we focus on the three basic pillars of security known as the CIA-Triad (confidentiality, integrity, and availability). Assume that an adversary wants to violate these security properties defined as follows [24]:

- *Confidentiality:* Unauthorised disclosure and misuse of the resources or information should be prevented.
- *Integrity:* Unauthorised changes, destruction, or loss of resources or information should be prevented.
- *Availability:* Authorised users should have timely and uninterrupted access to requested resources or information.

3.3. Mathematical Model of an Adversary

Based on the characteristics of a potential adversary introduced in Section 3.1, the generalised mathematical adversary model is formulated.

Definition 1. *The adversary characteristics described in Table 1 are denoted as:*

$$\mathcal{RS} \in \{Ex, Ex^{+}, In\},$$
$$\mathcal{CC} \in \{N, Y\}, \text{ and}$$
$$\mathcal{T} \in \{C, As, It\},$$

Then, the general prototype of an individual adversary $\mathcal{A}$ is defined by a triple of features as

$$\mathcal{A} = (\mathcal{RS}, \mathcal{CC}, \mathcal{T}).$$

While attacking a system, the adversary can obtain some information depending on their characteristics and the target of the attack. To formally describe this process, the following oracles are introduced, each representing a specific source of information that may be available to the adversary:

Oracle `Del`: The delegation oracle allows the adversary to interact with the delegate $\mathcal{D}$:

$$\{\mathsf{t}, \perp\} \leftarrow \mathtt{Del}(AD).$$

As input, `Del` receives the set of data AD that varies depending on the target of the adversary and application domain. It may include a party or resource id, etc. The oracle either returns an access token t or rejects the request by returning $\perp$.

Oracle `Cor`: The corruption oracle allows the adversary to corrupt parties and learn their secrets s:

$$\{\mathsf{s}, \perp\} \leftarrow \mathtt{Cor}(\mathsf{id}).$$

As input, `Cor` receives an identifier of a party id that the adversary wants to corrupt. The oracle will then return the secrets s belonging to the party associated with this id or $\perp$, e.g., if no such party exists.

Oracle `Obs`: The observe oracle allows the adversary to observe the communication between system parties and the delegate $\mathcal{D}$:

$$\{\mathsf{ts}, \perp\} \leftarrow \mathtt{Obs}(\mathsf{id}).$$

As input, `Obs` receives id, indicating which party is communicating with $\mathcal{D}$. The oracle will then return the transcript ts of the communication or $\perp$, e.g., if no party associated with id exists.

Oracle `ThP`: This oracle allows the adversary to interact with the third party $\mathcal{TP}$:

$$\{\mathsf{t}, \perp\} \leftarrow \mathtt{ThP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i}, \mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i})).$$

As input, `ThP` receives id, indicating which party delegates its rights, $\mathsf{id}_{\mathcal{D}}$ of the party which requested the rights, an identifier of desired resource i and the proof of rights $\mathcal{RP}$ based on id, $\mathsf{id}_{\mathcal{D}}$ and i. The oracle will either return an access token t or reject the request by returning $\perp$.

The adversary can query one or more oracles to obtain information depending on its characteristics and needs. However, according to Table 1, the adversary with $\mathcal{CC} = N$ has no corruption capability; hence, they are not able to query `Cor`. Also, `Obs` cannot be invoked by the adversary with $\mathcal{RS} = Ex$ due to limited observation capabilities. As it was

mentioned in Section 3.1, the cartesian product of all characteristics allows us to generate 18 types of adversaries ($3 \times 3 \times 2 = 18$). Table 2 presents the available oracles that $\mathcal{A}$ can query, depending on their set of characteristics.

It can be observed that some adversary types can have access to the same oracles, raising the question of what differentiates them. For example, $\mathcal{A}_7$ and $\mathcal{A}_8$ can both query `Del`, `Obs`, and `ThP`, but neither can access `Cor`. Additionally, they have a similar set of characteristics, the only difference being the time parameter $\mathcal{T}$. This can affect which attacks can be executed by $\mathcal{A}$. For example, $\mathcal{A}_7$ is restricted to attacks that can be executed within a concrete time frame (C), whereas $\mathcal{A}_8$ can observe communications over an extended period, enabling the collection of information for more sophisticated future attacks. Thus, even when adversaries have access to the same oracles, differences in their characteristics can significantly influence their attack capabilities, which may also vary across targeted systems.

Table 2. Ability of different adversaries to query oracles.

$\mathcal{A}$	Characteristics			`Del`	`Cor`	`Obs`	`ThP`
	$\mathcal{RS}$	$\mathcal{CC}$	$\mathcal{T}$				
1	Ex	N	C	✓	✗	✗	✓
2	Ex	N	As	✓	✗	✗	✓
3	Ex	N	It	✓	✗	✗	✓
4	Ex	Y	C	✓	✓	✗	✓
5	Ex	Y	As	✓	✓	✗	✓
6	Ex	Y	It	✓	✓	✗	✓
7	Ex^+	N	C	✓	✗	✓	✓
8	Ex^+	N	As	✓	✗	✓	✓
9	Ex^+	N	It	✓	✗	✓	✓
10	Ex^+	Y	C	✓	✓	✓	✓
11	Ex^+	Y	As	✓	✓	✓	✓
12	Ex^+	Y	It	✓	✓	✓	✓
13	In	N	C	✓	✗	✓	✓
14	In	N	As	✓	✗	✓	✓
15	In	N	It	✓	✗	✓	✓
16	In	Y	C	✓	✓	✓	✓
17	In	Y	As	✓	✓	✓	✓
18	In	Y	It	✓	✓	✓	✓

✓—an adversary can query the oracle; ✗—an adversary cannot query the oracle.

Also, during an attack, the adversary $\mathcal{A}$ has a certain probability of success. If there are no specific data and conditions that can enhance this probability, then $\mathcal{A}$ can succeed with probability equal to some constant c. If some additional useful information is provided, this probability can increase by ϵ and become $c + \epsilon$. In other words, this ϵ is an advantage that is achieved by $\mathcal{A}$. For our study, the advantage is defined as follows:

Definition 2. *Let* Π *be an Access Control, Authentication, Rights or Computation delegation protocol, then the advantage of* $\mathcal{A}$ *is defined as*

$$Adv_{\Pi}^{\mathsf{G}}(\mathcal{A}) := \Pr[\mathsf{G}(\mathcal{A}) \to 1]$$
$$\textit{for } \mathsf{G} \in \{\mathsf{AC}, \mathsf{AD}, \mathsf{RD}, \mathsf{CD}\},$$

where Pr *is a probability to succeed,* AC *is an Access Control Delegation,* AD *is an Authentication Delegation,* RD *is a Rights Delegation, and* CD *is a Computation Delegation game (see Section 4).*

4. Formalisation of the Security Properties

In this section, mathematical formalisations of CIA security properties are introduced (see Section 3.2) for each application domain of delegation defined in Section 2 with respect to the adversary model from Section 3. All formalisations are represented as security games:

- AC is an Access Control/Authorisation delegation game.
- AD is an Authentication delegation game.
- RD is a Rights delegation game.
- CD is a Computation delegation game.

An adversary $\mathcal{A}$ is considered to win the security game if they successfully violate the targeted security property (confidentiality, integrity, or availability).

In order to be able to provide mathematical formalisations and clearly define the required elements, the following collections are introduced:

- $PL := \{\mathsf{id} = 0, 1, \ldots \mid \mathcal{P}_{\mathsf{id}}\}$ is a list of all system participants.
- $RL := \{\mathsf{i} = 0, 1, \ldots \mid \mathcal{R}_{\mathsf{i}}\}$ is a list of all system resources.
- $M \subseteq PL \times RL$ denotes the access matrix. More precisely, if $(\mathsf{id}, \mathsf{i}) \in M$ then party $\mathcal{P}_{\mathsf{id}}$ has access to resource $\mathcal{R}_{\mathsf{i}}$. It contains all system participants.
- $F \subseteq PL$ includes all $\mathcal{P}_{\mathsf{id}}$ which are accessible from outside of a system.
- $H := \{i = 0, 1, \ldots \mid \mathsf{h}_{\mathsf{i}}\}$ includes all elements of a system that are expected in case the system operates without deviations. It can be parties, passwords, messages, etc.
- $L := \{\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i} \in M \mid \mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i})\}$ specifies all valid proofs of rights $\mathcal{RP}$ related to an owner party $\mathcal{P}_{\mathsf{id}}$, a delegate $\mathcal{D}$ with $\mathsf{id}_{\mathcal{D}}$ and a resource $\mathcal{R}_{\mathsf{i}}$ for each party involved in a system.
- $S := \{\mathsf{id} \in M \mid \mathsf{s}_{\mathsf{id}}\}$ contains all secrets of each party in a system.

In all of the models described below, one of the initial steps for an adversary is to receive an access matrix M as input. However, just as in the case of oracles, some adversaries, depending on their characteristics, may not receive M, for example, an adversary with $\mathcal{RS} = Ex$ receives only F.

4.1. Access Control and/or Authorisation Delegation

Assume that $\exists\, \mathsf{id}, \mathsf{i} : (\mathsf{id}, \mathsf{i}) \in M$, then, in the considered application domain, a third party element $\mathcal{TP}_{\mathsf{id}}$ requests from a delegate $\mathcal{D}$ access to a resource $\mathcal{R}_{\mathsf{i}}$ provided by an RC system. $\mathcal{D}$ replies with a token t. Using t, $\mathcal{TP}_{\mathsf{id}}$ can access desired $\mathcal{R}_{\mathsf{i}}$.

The task of an adversary $\mathcal{A}$ is to disrupt the normal operation of the system and achieve their goal of violating one or more CIA properties. To model this process, a security game described in Definition 3 is introduced.

Definition 3. *Let* Π *be an access control or authorisation protocol, and* AC *be the following experiment:*

1. *Initializing M, F, H, S.*
2. $\mathcal{A}$ *receives as input the access matrix M or F.*
3. $\mathcal{A}$ *is assigned identifier* $\mathsf{id}_{\mathcal{A}}$.

4. $\mathcal{A}$ *can query the following oracles:*

$$\{\mathsf{t}, \perp\} \leftarrow \mathtt{Del}(\mathsf{id}, \mathsf{i}),$$
$$\{\mathsf{s}, \perp\} \leftarrow \mathtt{Cor}(\mathsf{id}), \textit{ and}$$
$$\{\mathsf{ts}, \perp\} \leftarrow \mathtt{Obs}(\mathsf{id}),$$

where id *is an identifier of a protocol party* $\mathcal{TP}_{\mathsf{id}}$ *which wants to obtain access to an RC system resource* $\mathcal{R}_{\mathsf{i}}$.

5. $\mathcal{A}$ *wins if any of the following cases hold:*
   - *Confidentiality:* $\mathcal{A}$ *outputs a token* t *and a resource identifier* i*, such that*
     (1) $\mathcal{A}$ *is not allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M*, but*
     (2) $\mathcal{D}$ *grants* $\mathcal{A}$ *access to* $\mathcal{R}_{\mathsf{i}}$ *when provided with* t.
   - *Integrity:* $\mathcal{A}$ *outputs a party identifier* id*, a resource identifier* i*, and a modified item* $\mathsf{h}_{\mathcal{A}}$ *such that*
     (1) there is a third party $\mathcal{TP}_{\mathsf{id}}$ *that is allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M*, and*
     (2) $\mathcal{A}$ *is not allowed to access to* $\mathcal{R}_{\mathsf{i}}$ *according to* M*, but*
     (3) $\exists \mathsf{h}_{\mathcal{A}}$ *that does not match its corresponding item* $\mathsf{h} \in H$.
   - *Availability:* $\mathcal{A}$ *outputs an identifier* id *indicating party* $\mathcal{TP}_{\mathsf{id}}$ *and a resource identifier* i*, such that*
     (1) $\mathcal{TP}_{\mathsf{id}}$ *has access to* $\mathcal{R}_{\mathsf{i}}$ *according to* M*, but*
     (2) access using a valid token t *is not granted.*

4.2. Authentication Delegation

Assume that $\exists\, \mathsf{id}, \mathsf{i} : (\mathsf{id}, \mathsf{i}) \in M$, then, in this application domain, a party $\mathcal{P}_{\mathsf{id}}$ included in an RC system requests from a delegate $\mathcal{D}$ to authenticate against a third party $\mathcal{TP}$ on behalf of $\mathcal{P}_{\mathsf{id}}$, in order to $\mathcal{P}_{\mathsf{id}}$ grants access to $\mathcal{R}_{\mathsf{i}}$. In case of successful authentication, $\mathcal{TP}$ sends to $\mathcal{D}$ a token t such that $\mathcal{P}_{\mathsf{id}}$ is allowed to access desired $\mathcal{R}_{\mathsf{i}}$.

An adversary $\mathcal{A}$ targets the system elements in order to violate CIA properties. To model this process, a security game described in Definition 4 is introduced.

Definition 4. *Let* Π *be an authentication delegation protocol, and* AD *be the following experiment:*

1. *Initialising* M*,* F*,* H*,* S.
2. $\mathcal{A}$ *receives as input the access matrix* M *or* F.
3. $\mathcal{A}$ *is assigned identifier* $\mathsf{id}_{\mathcal{A}}$.
4. $\mathcal{A}$ *can query the following oracles:*

$$\{\mathsf{t}, \perp\} \leftarrow \mathtt{Del}(\mathsf{id}, \mathsf{i}),$$
$$\{\mathsf{s}, \perp\} \leftarrow \mathtt{Cor}(\mathsf{id}), \textit{ and}$$
$$\{\mathsf{ts}, \perp\} \leftarrow \mathtt{Obs}(\mathsf{id}),$$

where id *identifies an RC system party* $\mathcal{P}_{\mathsf{id}}$ *which wants to be authenticated to interact with resource* $\mathcal{R}_{\mathsf{i}}$ *of the third party* $\mathcal{TP}$.

5. $\mathcal{A}$ *wins if any of the following cases hold:*
   - *Confidentiality:* $\mathcal{A}$ *outputs a resource identifier* i *and token* t*, such that*
     (1) $\mathcal{A}$ *is not allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M*, but*
     (2) $\mathcal{TP}$ *grants* $\mathcal{A}$ *access to* $\mathcal{R}_{\mathsf{i}}$ *when provided with* t.
   - *Integrity:* $\mathcal{A}$ *outputs a party identifier* id*, a resource identifier* i*, a token* t*, and a modified item* $\mathsf{h}_{\mathcal{A}}$ *such that*

(1) $\mathcal{P}_{\mathsf{id}}$ *is allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M *with provided token* t, *and*
(2) $\mathcal{A}$ *is not allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M, *but*
(3) $\exists \mathsf{h}_{\mathcal{A}}$ *which does not match corresponding item* $\mathsf{h} \in H$.

- *Availability:* $\mathcal{A}$ *outputs a party identifier* id *and a resource identifier* i, *such that*
  (1) $\mathcal{P}_{\mathsf{id}}$ *is allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M, *but*
  (2) $\mathcal{TP}$ *does not grants* $\mathcal{P}_{\mathsf{id}}$ *access to* $\mathcal{R}_{\mathsf{i}}$ *when provided with* t.

4.3. Rights Delegation

Assume that $\exists\, \mathsf{id}, \mathsf{i} : (\mathsf{id}, \mathsf{i}) \in M$ and $\exists\, \mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i}) \in L$, then, according to right delegation scenario, a party $\mathcal{P}_{\mathsf{id}}$ (RC system) possesses some kind of rights related to a resource $\mathcal{R}_{\mathsf{i}}$ owned by a third party $\mathcal{TP}$. A delegate $\mathcal{D}$ wants to access $\mathcal{R}_{\mathsf{i}}$ and in order to be able to do it, it requests the rights from $\mathcal{P}_{\mathsf{id}}$. In turn, $\mathcal{P}_{\mathsf{id}}$ sends a $\mathcal{RP} = \mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i})$ to $\mathcal{D}$, so it can request access to $\mathcal{R}_{\mathsf{i}}$ from $\mathcal{TP}$ and interact with it on its own.

The goal of an adversary $\mathcal{A}$ is to win the security game described in Definition 5, such that winning ensures that the confidentiality, integrity, and/or availability properties are violated.

Definition 5. *Let* Π *be a rights delegation protocol, and* RD *be the following experiment:*

1. *Initialising* M, F, H, L, S.
2. $\mathcal{A}$ *receives as input the access matrix* M *or* F.
3. $\mathcal{A}$ *is assigned identifier* $\mathsf{id}_{\mathcal{A}}$.
4. $\mathcal{A}$ *can query the following oracles:*

$$\{\mathsf{t}, \perp\} \leftarrow \mathtt{ThP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i}, \mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i})),$$
$$\{\mathsf{s}, \perp\} \leftarrow \mathtt{Cor}(\mathsf{id}), \text{ and}$$
$$\{\mathsf{ts}, \perp\} \leftarrow \mathtt{Obs}(\mathsf{id}),$$

   where id *identifies an RC system party* $\mathcal{P}_{\mathsf{id}}$ *which transfers its rights related to resource* $\mathcal{R}_{\mathsf{i}}$ *owned by* $\mathcal{TP}$, $\mathsf{id}_{\mathcal{D}}$ *identifies a delegate* $\mathcal{D}$ *which obtains rights by possessing proof of rights* $\mathcal{RP}$ *from* $\mathcal{P}_{\mathsf{id}}$.
5. $\mathcal{A}$ *wins if any of the following cases hold:*
   - *Confidentiality:* $\mathcal{A}$ *outputs a resource identifier* i *and a token* t, *such that*
     (1) $\mathcal{A}$ *is not allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M, *and*
     (2) $\mathcal{A}$ *with* $\mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{A}}, \mathsf{i})$ *is not allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* L, *but*
     (3) $\mathcal{TP}$ *grants* $\mathcal{A}$ *access to* $\mathcal{R}_{\mathsf{i}}$ *when provided with* t *and* $\mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{A}}, \mathsf{i})$.
   - *Integrity:* $\mathcal{A}$ *outputs a resource identifier* i, *token* t, *and a modified item* $\mathsf{h}_{\mathcal{A}}$ *such that*
     (1) $\mathcal{P}_{\mathsf{id}}$ *has access to* $\mathcal{R}_{\mathsf{i}}$, *and*
     (2) $\mathcal{A}$ *is not allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M, *and*
     (3) $\mathcal{A}$ *with* $\mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{A}}, \mathsf{i})$ *is not allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* L, *but*
     (4) $\exists \mathsf{h}_{\mathcal{A}}$ *does not match corresponding item* $\mathsf{h} \in H$.
   - *Availability:* $\mathcal{A}$ *outputs a resource identifier* i *and token* t, *such that*
     (1) $\mathcal{P}_{\mathsf{id}}$ *is allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* M, *and*
     (2) $\mathcal{D}$ *with* $\mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i})$ *is allowed to access* $\mathcal{R}_{\mathsf{i}}$ *according to* L, *but*
     (3) $\mathcal{TP}$ *does not grants* $\mathcal{D}$ *access to* $\mathcal{R}_{\mathsf{i}}$ *when provided with* t *and* $\mathcal{RP}(\mathsf{id}, \mathsf{id}_{\mathcal{D}}, \mathsf{i})$.

4.4. Computation Delegation

Assume that $\mathsf{id} \in M$, then in the discussed application domain of delegation, an RC system $\mathcal{P}_{\mathsf{id}}$ delegates costly computations m to a delegate $\mathcal{D}$. $\mathcal{D}$ processes the request and replies with m^*.

An adversary $\mathcal{A}$ tries to obtain secrets of $\mathcal{P}_{\mathsf{id}}$ or $\mathcal{D}$ in order to win the security game.

Definition 6. *Let* Π *be a Computation delegation protocol, and* CD *be the following experiment:*

1. *Initialising M, F, H, S.*
2. $\mathcal{A}$ *receives as input the access matrix M or F.*
3. $\mathcal{A}$ *is assigned identifier* $\mathsf{id}_{\mathcal{A}}$.
4. $\mathcal{A}$ *can query the following oracles:*

$$\{\mathsf{m}^*, \perp\} \leftarrow \texttt{Del}(\mathsf{id}, \mathsf{m}),$$
$$\{\mathsf{s}, \perp\} \leftarrow \texttt{Cor}(\mathsf{id}), \textit{ and}$$
$$\{\mathsf{ts}, \perp\} \leftarrow \texttt{Obs}(\mathsf{id}),$$

   where id *identifies an RC system party* $\mathcal{P}_{\mathsf{id}}$, m *is an unprocessed message and* m^* *is a processed message by a delegate* $\mathcal{D}$*, e.g., encrypted or decrypted.*
5. $\mathcal{A}$ *wins if any of the following cases hold:*
   - *Confidentiality:* $\mathcal{A}$ *outputs a party identifier* id *and a secret* s*, such that*
     (1) $\mathcal{A}$ *is not* $\mathcal{P}_{\mathsf{id}}$ *requested* $\mathcal{D}$ *(*$\mathsf{id}_{\mathcal{A}} \neq \mathsf{id}$*), but*
     (2) s *belongs to* $\mathcal{P}_{\mathsf{id}}$.
   - *Integrity:* $\mathcal{A}$ *outputs a party identifier* id*, a secret* s *and a modified item* $\mathsf{h}_{\mathcal{A}}$ *such that*
     (1) $\mathcal{P}_{\mathsf{id}}$ *legitimately belongs to the interaction process, and*
     (2) s *belongs to* $\mathcal{P}_{\mathsf{id}}$*, but*
     (3) $\exists \mathsf{h}_{\mathcal{A}}$ *does not match corresponding item* $\mathsf{h} \in H$.
   - *Availability:* $\mathcal{A}$ *outputs a party identifier* id *and a secret* s*, such that*
     (1) $\mathcal{P}_{\mathsf{id}}$ *legitimately belongs to the interaction process, and*
     (2) s *belongs to* $\mathcal{P}_{\mathsf{id}}$*, but*
     (3) $\mathcal{P}_{\mathsf{id}}$ *obtains either no result for its request or an incorrect one.*

5. Security Evaluation of Existing Protocols

In particular, we consider investigating whether the delegation approaches identified in the literature (Appendix A) can be shown to be secure under the formed definitions of Section 4. In this section, the robustness of these delegation solutions is qualitatively assessed against three types of adversaries aiming to violate confidentiality, integrity, and availability.

5.1. Adversary Types

As noted earlier, 18 types of adversaries (see Section 3.3) can be defined by combining the possible characteristics listed in Table 1. In order to provide a detailed illustrative example, we focus only on the three most representative and distinct types of adversaries:

$\mathcal{A}_{weak}$: An adversary belonging to this type is a "weak" one. They are external, have no corruption capability, and are able to perform attacks feasible within a concrete (typically short) time interval. $\mathcal{A}_{weak}$ can query the following oracles: `Del`, `ThP` (see Table 2 (1)).

$\mathcal{A}_{middle}$: Adversaries in this group can be described as "medium" ones. They have External+ access to the system; hence, they can observe the communication between

parties but have no corruption capability. However, $\mathcal{A}_{middle}$ may have accomplices within the system who may have a corruption capability. Their position allows more sophisticated attacks than $\mathcal{A}_{weak}$. $\mathcal{A}_{middle}$ can query the following oracles: `Del`, `Obs`, `ThP` (see Table 2 (8)).

$\mathcal{A}_{strong}$: When an adversary is categorised as "strong", it is assumed that they have direct access to the system, acting as insiders with full corruption capability. Combined with unlimited time, this enables them to carry out the most resource-intensive attacks. $\mathcal{A}_{strong}$ can query the following oracles: `Del`, `Cor`, `Obs`, `ThP` (see Table 2 (18)).

5.2. Process and Results of Security Evaluation

In this section, the approaches presented in the papers listed in Table A1 are evaluated to determine whether they are likely to be secure against the adversaries introduced above. Due to the large number of publications, we are not able to provide full formal proofs or attacks for each individual paper. Rather, we give our perspective on whether a construction could be proven secure with some effort or whether attacks exist. Thus, later in this section, a detailed analysis of one approach from each application domain is shown. A summary of the analysis for each paper included in this SoK is provided in Table A2.

5.2.1. Access Control and/or Authorisation Delegation

In this domain, the group-oriented access control security mechanism based on DTLS presented by Park and Park (see Section 2.1, Figure 3) is examined. According to the Definition 3, $\mathcal{A}_{weak}$ initially receives as an input list F identified parties which are remotely reachable by $\mathcal{A}_{weak}$. $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ receive as input the access matrix M; hence, they are aware of all participants of the system, and their legitimate access.

In this application domain, `ThP` is not available; therefore, it is assumed that $\mathcal{A}_{weak}$ can query only `Del` and interact directly with $\mathcal{D}$. In order to violate the confidentiality and integrity of any system participant $\mathcal{TP}$, $\mathcal{A}_{weak}$ has to obtain its secret `s` and perform authorisation pretending to be $\mathcal{TP}$. However, as $\mathcal{A}_{weak}$ does not have the corruption capability, they cannot defraud the target client acting as $\mathcal{TP}$ to obtain its `s` and alter the associated resources. Hence, the given system can be proven secure against $\mathcal{A}_{weak}$ in terms of confidentiality and integrity. Considering the availability of the system, $\mathcal{A}_{weak}$ is able to disrupt system functionality by exploiting the single point of failure (SPoF) vulnerability via performing a successful denial of service (DoS) attack against the SP acting as $\mathcal{D}$, sending multiple access requests.

In addition to `Del`, $\mathcal{A}_{middle}$ can query `Obs`. This allows $\mathcal{A}_{middle}$ to eavesdrop on the communication; however, due to using DTLS, confidentiality remains secure. However, having Ex^{+} position relating to the system, $\mathcal{A}_{middle}$ is potentially able to modify data from sensor devices using allies illegally. Also, by performing DoS attacks against SP, $\mathcal{A}_{middle}$ can violate system availability. Therefore, it can be concluded that this system is not secure against $\mathcal{A}_{middle}$ regarding confidentiality and availability properties, but secure concerning integrity.

$\mathcal{A}_{strong}$ is an insider with access to all available oracles and unlimited time. Thus, they potentially can retrieve secrets `s` of any system participant (e.g., security proxy, clients), alter any data from sensor devices, and disrupt the availability of these parties. Therefore, the given system is not considered secure against this type of adversary in terms of confidentiality, integrity, and availability.

5.2.2. Authentication Delegation

In this application domain, the D2TLS protocol introduced in [22] (see Section 2.2, Figure 4) is examined. According to Definition 4, $\mathcal{A}_{weak}$ initially receives as an input list

F identified parties which are remotely reachable by $\mathcal{A}_{weak}$. $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ receive as input the access matrix M; hence, they are aware of all participants of the system, and their legitimate access.

In this application domain, $\mathcal{A}_{weak}$ is able to query only `Del`. As $\mathcal{A}_{weak}$ does not have corruption capability, they cannot compromise any protocol participants in order to obtain their `s`. Hence, $\mathcal{A}_{weak}$ cannot be authenticated instead of legitimate clients and tamper with their resources $\mathcal{R}$. Based on this, the protocol can be proven secure against $\mathcal{A}_{weak}$ in terms of confidentiality and integrity. Considering the capability of $\mathcal{A}_{weak}$ to violate the availability property, $\mathcal{A}_{weak}$ is able to perform DoS attacks against the SA acting as $\mathcal{D}$ by sending numerous delegation requests. Additionally, DoS attacks against a client during protocol operation can cause its interruption. Therefore, this protocol is not secure against $\mathcal{A}_{weak}$ regarding availability.

$\mathcal{A}_{middle}$ can query not only `Del` but also `Obs`. In the given protocol, in order to be successfully authenticated, the adversary should not only have credentials but also be able to forge the client's signature during run time. We assume that this is likely not feasible even within the asymptotic time available for $\mathcal{A}_{middle}$, and the given protocol is secure against $\mathcal{A}_{middle}$ in terms of confidentiality. In this scenario, if confidentiality is not violated, integrity is not violated either. Attacks on communication channels, such as man-in-the-middle (MitM), also cannot be successful because the presented protocol employs the DTLS protocol, which is based on the TLS protocol that ensures the confidentiality and integrity of information during transmission. Therefore, this protocol is likely secure against $\mathcal{A}_{middle}$ in terms of integrity. Availability of the system can be disrupted by performing DoS attacks against SA and clients.

As the most powerful adversary, insider $\mathcal{A}_{strong}$ has access to all available oracles (`Del`, `Cor`, `Obs`) and also has unlimited time It. These resources allow $\mathcal{A}_{strong}$ to perform spoofing, information disclosure, tampering, repudiation, elevation of privilege, and DoS attacks against clients and the SA. Having enough time, $\mathcal{A}_{strong}$ can spoof a legitimate client and forge its signature in order to perform the authentication process. Also, $\mathcal{A}_{strong}$ can compromise the SA and manipulate its decisions, violating the confidentiality and integrity of the data and the availability of the system.

5.2.3. Rights Delegation

In this domain, the delegation of rights in IoT based on Blockchain technology [23] (see Section 2.3, Figure 5) is discussed. Following Definition 5, $\mathcal{A}_{weak}$ initially receives as an input list F identified parties which are remotely reachable by $\mathcal{A}_{weak}$. $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ receive as input the access matrix M; hence, they are aware of all participants of the system, and their legitimate access. Also, they all can query `ThP`. In this application domain, $\mathcal{A}_{weak}$ has access to two oracles (`Del`, `ThP`). The absence of corruption capability leads to the impossibility of $\mathcal{A}_{weak}$ to obtain any secrets. For example, an attempt to request $\mathcal{RP}$ for $\mathcal{R}$ cannot be successful for unauthenticated $\mathcal{A}_{weak}$, because of recurring validations within the blockchain. Therefore, if $\mathcal{A}_{weak}$ queries the resource manager acting as $\mathcal{TP}$ via `ThP` without a valid $\mathcal{RP}$, `ThP` will return $\perp$. Also, $\mathcal{A}_{weak}$ cannot compromise parties and manipulate available data; hence, the integrity of the data is ensured. In order to violate the availability of the system, $\mathcal{A}_{weak}$ can perform DoS attacks against the resource manager by requesting access to $\mathcal{R}$; however, first, recurring validation of a user prevents illegitimate items from making several requests; second, suspicious amount of requests leads to restrictions of suspicious items. DoS attacks against the buyer acting as $\mathcal{D}$ do not influence system functionality, since $\mathcal{A}_{weak}$ is only able to run attacks during C time, so the buyer can continue its operations after a DoS attack is finished. Summarising the above, it

can be concluded that this system is completely secure from $\mathcal{A}_{weak}$ with respect to all three security goals.

Since $\mathcal{A}_{middle}$ can query `Obs` and hence can observe communication. However, a blockchain-based network and smart contracts prevent the violation of data confidentiality and integrity. Considering the Ex^{+} relation to the system of $\mathcal{A}_{middle}$, they can have accomplices within the system, e.g., the resource manager. Hence, $\mathcal{A}_{middle}$ can compromise resource $\mathcal{R}$ and retrieve some information about the buyer or broker or tamper with their data. Also, corruption of the resource manager and its $\mathcal{R}$ leads to possible incorrect system behaviour, when the legitimate buyer with valid $\mathcal{RP}$ does not have access to the desired $\mathcal{R}$, for instance, if the list of approved users is modified. Therefore, this system is not secure against $\mathcal{A}_{middle}$ regarding all CIA-Triad.

Despite $\mathcal{A}_{strong}$ has more resources and capabilities than $\mathcal{A}_{weak}$ and $\mathcal{A}_{middle}$, it does not give $\mathcal{A}_{strong}$ a significant advantage in the results they can achieve by attacking this system in comparison with others. In order to obtain or manipulate information during the transfer of the right, $\mathcal{A}_{strong}$ should compromise the majority of nodes, which we assume is not possible even considering available It time. Therefore, $\mathcal{A}_{strong}$ can perform similar attacks to $\mathcal{A}_{middle}$, and the system is secure against $\mathcal{A}_{strong}$ with respect to all three security properties.

5.2.4. Computation Delegation

In this domain, the application of our mathematical model on the example of Lightweight Revocable Hierarchical Attribute-Based Encryption for the IoT [15] (see Section 2.4, Figure 6) is presented. Based on Definition 6, $\mathcal{A}_{weak}$ initially receives as an input list F identified parties which are remotely reachable by $\mathcal{A}_{weak}$. $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ receive as input the access matrix M; hence, they are aware of all participants of the system, and their legitimate access. As `ThP` is not available for this application domain, $\mathcal{A}_{weak}$ can query only `Del`. The absence of the capability of corruption does not allow this kind of adversary to acquire any secrets `s` of the system parties and to tamper with information within the system. However, $\mathcal{A}_{weak}$ can perform DoS attacks against the devices of data owners. Hence, the given system can be proven secure against $\mathcal{A}_{weak}$ in terms of confidentiality and integrity, but insecure in terms of availability.

The authors assume that all system elements are trusted and cannot be compromised except for data users, but even they are not able to disclose information and tamper with data. Additionally, all communication channels between system parties are considered secure against confidentiality and integrity violations, except three channels (between domain authorities and data owners, the cloud service provider and data owners, and the cloud service provider and data users) that can be susceptible to eavesdropping. Hence, the adversaries $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ can violate the confidentiality of the system by listening to these three channels, but they are not able to perform attacks against the integrity of the system data. Considering the possibilities of disrupting system availability, we see the potential for $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ to perform DoS attacks against the cloud service provider, causing SPoF. Concluding the above, the given system is not secure against attacks from $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ aimed at the availability property.

6. Discussion

During the analysis of the papers selected for our SoK, the approaches proposed by the authors were evaluated in terms of their resilience to attacks carried out by adversaries with varying capabilities, targeting the fundamental security properties of the system.

Given the extensive number of existing approaches, we provided a detailed security analysis for only four representative ones, one from each application domain (see Section 5).

The remaining evaluations were conducted using the same methodology. A summary of the results is presented in Table A2 and discussed below.

6.1. Access Control and/or Authorisation Delegation

In this domain, all approaches focus on delegating decision-making functions. Specifically, the RC system delegates access control and authorisation to a delegate, which then determines whether a third party is granted access. Most approaches either delegate these functions to a single device or utilise blockchain as a delegate. Additionally, some delegates incorporate monitoring mechanisms to track the activity of third parties and respond accordingly. For secure data transmission, the TLS protocol is predominantly employed.

The overall architecture of protocols related to this domain is illustrated in Figure 2a, and the summarised results of their security evaluation are presented in Figure 7a.

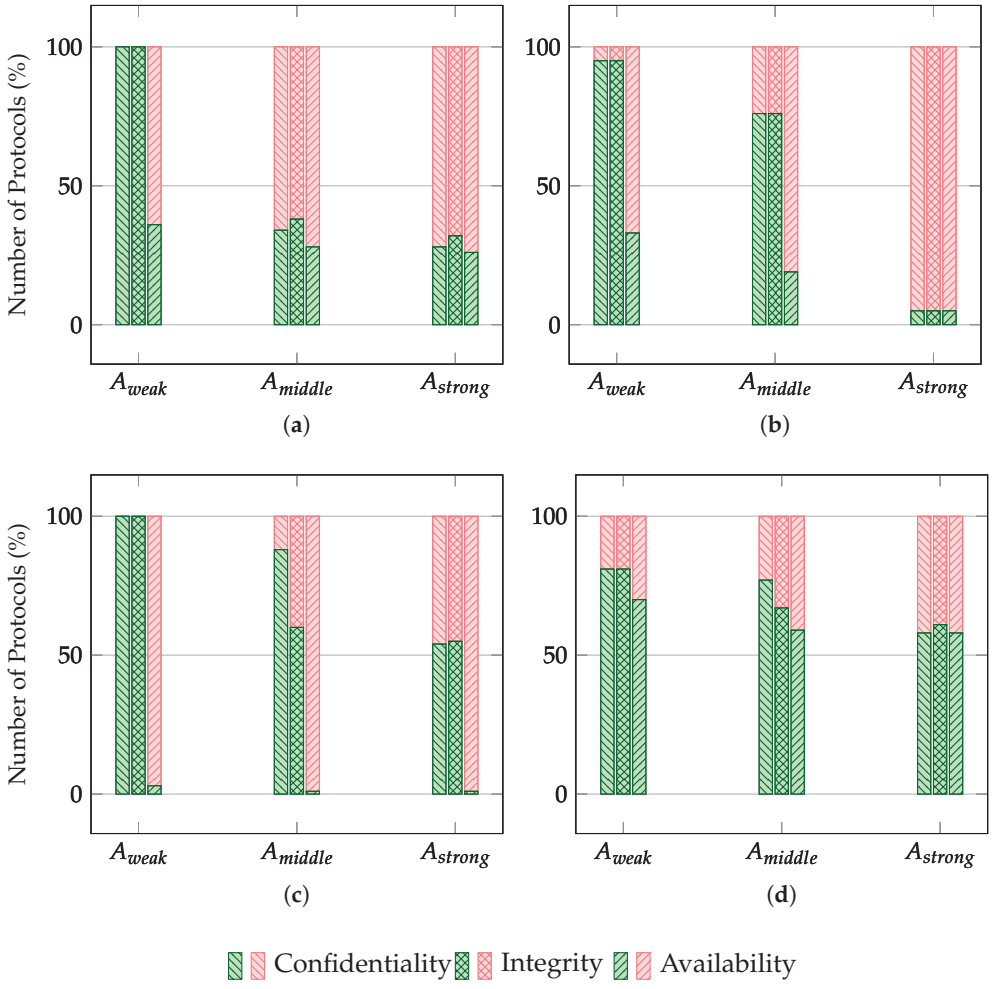


Figure 7. Number of secure (green) and insecure (red) delegation protocols against three types of adversaries in terms of violating confidentiality, integrity, and availability for (**a**) Access Control and/or Authorisation, (**b**) Authentication, (**c**) Rights, and (**d**) Computation delegation application domains.

6.1.1. Confidentiality

It can be observed that all systems in this domain are likely protected against $\mathcal{A}_{weak}$ in terms of confidentiality. $\mathcal{A}_{weak}$ can only observe communications, but cannot corrupt system parties and subsequently is unable to extract secrets. Moreover, many protocols include strong security measures such as encryption, key management, secure transport layers (e.g., DTLS [21,25]), and decentralised architectures (e.g., blockchain [17,26,27]) to protect sensitive data.

When $\mathcal{A}_{middle}$ is considered, a significantly smaller number of protocols are classified as secure. Although $\mathcal{A}_{middle}$, similar to $\mathcal{A}_{weak}$, lacks direct corruption capabilities, they may have accomplices within the system who assist them in obtaining sensitive data. These internal allies may potentially exploit vulnerabilities in access control mechanisms or gain direct access to cryptographic keys or secret information. For example, some protocols [25,28–30] are vulnerable to such internal threats, because $\mathcal{A}_{middle}$'s accomplices with legitimate access privileges may bypass protection layers or misuse their access to retrieve encrypted data. In contrast, blockchain-based protocols [26,31–33], which rely on immutable ledgers, provide stronger confidentiality guarantees in the presence of internal threats. They use cryptographic validation and distributed consensus, making it significantly more difficult for an insider to manipulate or access sensitive data undetected.

As $\mathcal{A}_{strong}$ is assumed to be a powerful adversary with extended access to the system and the ability to corrupt or manipulate critical system components, it is expected that more than half of the protocols under consideration are vulnerable. Similar to $\mathcal{A}_{middle}$, most of the affected protocols rely on traditional access control mechanisms, which are more susceptible to insider threats and compromise. In contrast, blockchain-based protocols demonstrate greater resistance due to their decentralised architecture, which makes it significantly more difficult to bypass or manipulate system components, even by insiders.

6.1.2. Integrity

Since $\mathcal{A}_{weak}$ does not have the ability to corrupt system components, the integrity of the data in all protocols remains unaffected.

For $\mathcal{A}_{middle}$, integrity is compromised in many protocols due to internal accomplices who can bypass access controls and potentially tamper with data. Although these protocols rely on encryption, digital signatures, and access control mechanisms, $\mathcal{A}_{middle}$'s accomplices may manipulate these mechanisms because they have privileged access to the system, making such protocols vulnerable to internal tampering. For example, while OAuth 2.0 and DTLS can guarantee integrity against external attacks from A_{weak}, $\mathcal{A}_{middle}$'s accomplices within the system may alter or corrupt data without detection. However, the use of blockchain in protocols prevents even internal accomplices from undetected data alteration, since each modification must be validated by the network, preventing tampering without consensus.

Similar conclusions can be drawn regarding the influence of $\mathcal{A}_{strong}$ on protocol integrity as with $\mathcal{A}_{middle}$. However, since $\mathcal{A}_{strong}$ is more powerful and has broader access to the system, even fewer protocols remain protected. Nevertheless, blockchain-based protocols continue to offer strong resistance. To compromise the integrity of such systems, $\mathcal{A}_{strong}$ would need to control more than 50% of the network, which is highly unlikely. Therefore, blockchain protocols maintain a high level of integrity assurance, even against strong adversaries.

It is logical to assume that, for any adversary, the number of systems protected against confidentiality and integrity violations should be equal, as in the case of $\mathcal{A}_{weak}$. However, for $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$, there are more systems that appear to maintain integrity than confidentiality. The reason for this is that some delegation solutions [32,34] include monitoring mechanisms capable of detecting suspicious activity or data manipulation and responding accordingly. For example, suppose an adversary may gain unauthorised access to data, thus violating confidentiality, without triggering any alarms, as long as the data remains unmodified. However, if the adversary attempts to alter the data, thereby violating integrity, the system's monitoring mechanisms are likely to detect and react to the intrusion.

6.1.3. Availability

When it comes to availability, it is reasonable to assert that the stronger the adversary, the fewer systems are capable of withstanding their attacks. $\mathcal{A}_{weak}$ can succeed primarily by targeting peripheral components of the system, such as users or delegates, through attacks (e.g., DoS) which can be executed from an external position. In many protocols, such attacks do not result in critical system failures and are therefore still considered secure in terms of availability. However, for example, in the healthcare domain [35,36], even minor disruptions of the operation of patient wearable devices, e.g., pulsimeters, can lead to significant damage. In such cases, the system's primary objective is to protect the individual user rather than just the system itself, and therefore, these protocols are considered vulnerable.

In contrast, $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$ are capable of disrupting the operation of critical system components, potentially leading to significant availability issues. However, despite their extensive resources and capabilities, the impact of such adversaries can be mitigated in protocols that incorporate decentralisation and redundancy into the system architecture [33,37]. These design choices enhance fault tolerance and ensure that the system can continue functioning even when individual components are compromised.

6.2. Authentication Delegation

In this domain, all approaches delegate the authentication function to a delegate, which performs the authentication process on behalf of the RC system with respect to the third party. These solutions primarily rely on existing technologies, such as OAuth, digital certificates, and DTLS handshakes, all of which are executed by the delegate.

The overall architecture of protocols related to this domain is illustrated in Figure 2b, and the summarised results of their security evaluation are presented in Figure 7b.

6.2.1. Confidentiality

Most of the protocols in this domain ensure confidentiality against $\mathcal{A}_{weak}$, as they use strong security techniques (e.g., DTLS [22,38], OAuth [39,40], certificates [41], zero-knowledge proofs [19]). Since $\mathcal{A}_{weak}$ can observe the communication but not corrupt it, using proper encryption and secure authentication mechanisms protects the transmitted data.

Considering security against $\mathcal{A}_{middle}$, confidentiality is compromised in several protocols due to $\mathcal{A}_{middle}$'s ability to exploit internal vulnerabilities or weak access controls [39,40,42–44]. Internal accomplices may potentially bypass encryption mechanisms or gain unauthorised access to sensitive data. However, most protocols that include, e.g., DTLS [22,38] or certificates [41] are generally secure against $\mathcal{A}_{middle}$ in terms of confidentiality. These protocols rely on strong authentication mechanisms, secure session management, and encryption to ensure that even if the adversary can observe the communication, they cannot easily access the sensitive data even in the presence of internal threats.

For $\mathcal{A}_{strong}$, confidentiality is compromised across almost all the protocols discussed, as this type of internal adversary has extended control over the system. They can manipulate access control, corrupt encryption keys, or modify communication channels. Because $\mathcal{A}_{strong}$ has access to internal resources, they can bypass most security mechanisms that are effective against external threats. Nevertheless, protocols that employ a decentralised system architecture [45,46] can maintain confidentiality even against such powerful adversaries.

6.2.2. Integrity

In this domain, the number of systems secure in terms of confidentiality is equal to those secure in terms of integrity across all types of adversaries. It can be associated with the absence of any monitoring mechanisms in the evaluated systems. As a result, a violation of confidentiality typically leads to a violation of integrity. For example, if $\mathcal{A}$ manages to capture the user's credentials and authenticate on their behalf to gain access to resources, $\mathcal{A}$ can often also manipulate those resources by impersonating the user. The same logic applies in reverse. If $\mathcal{A}$ cannot illegally authenticate and gain access to resources intended for the user (i.e., cannot violate confidentiality), then they cannot manipulate these resources to compromise integrity either.

Considering the influence of $\mathcal{A}_{weak}$ on system integrity, it can be concluded that $\mathcal{A}_{weak}$ is not able to violate it, as they can only observe the communication without the capability to alter or corrupt it.

Against $\mathcal{A}_{middle}$, integrity remains secure for most of the protocols because they use digital signatures, message authentication codes and hashes to ensure that any unauthorised modifications to data can be detected [41,47,48]. However, some protocols [39,40,42,43] may be vulnerable, because internal $\mathcal{A}_{middle}$'s accomplices could have access to sensitive tokens, session data, or authentication credentials, allowing them to tamper with the data or compromise their integrity.

For $\mathcal{A}_{strong}$, integrity is compromised in most protocols, as this adversary can exploit internal access to directly tamper with or modify data. The ability to manipulate keys, authentication data, or communication channels from within makes it difficult for these protocols to ensure data integrity. However, protocols employing a decentralised system architecture [45,46] demonstrate resistance even against $\mathcal{A}_{strong}$ due to their distributed nature.

6.2.3. Availability

Many protocols in this domain are vulnerable to $\mathcal{A}_{weak}$ in terms of availability. These protocols primarily focus on ensuring confidentiality and integrity, but often lack mechanisms to address availability threats such as DoS attacks. Hence, $\mathcal{A}_{weak}$ can exploit this by launching DoS attacks, overloading system resources, and preventing legitimate users from accessing services [16,41,47,49].

There are even fewer protocols that ensure availability against $\mathcal{A}_{middle}$, as this adversary not only possesses the capabilities of $\mathcal{A}_{weak}$ but can also exploit internal accomplices, who may potentially bypass built-in protection mechanisms against availability threats, such as rate limiting or traffic filtering. The same applies to $\mathcal{A}_{strong}$, which as the most powerful adversary, can exploit both external and internal access. However, the use of decentralised system architectures [46] can effectively mitigate availability attacks, even from the strongest adversaries.

6.3. Rights Delegation

In this domain, RC systems transfer to a delegate rights to use, own, or access specific resources (third party). This is typically achieved either through blockchain technology, which ensures that only authorised participants can acquire these rights, or through public key cryptography, where access is granted via cryptographic keys, possession of which is considered de facto proof of right. The existing approaches often incorporate mechanisms applied to third parties, such as limiting the number of granted access rights or introducing resource redundancy. However, articles in this category rarely address the methods and protocols employed for secure information transmission.

The overall architecture of protocols related to this domain is illustrated in Figure 2c, and the summarised results of their security evaluation are presented in Figure 7c.

6.3.1. Confidentiality

All protocols presented in this domain are secure against $\mathcal{A}_{weak}$, as they incorporate encryption mechanisms that prevent $\mathcal{A}_{weak}$ from extracting any sensitive information through passive observation of the communication.

Most protocols are also secure against $\mathcal{A}_{middle}$. Despite the presence of internal accomplices, several solutions provide strong confidentiality guarantees. For instance, some rely on blockchain-based designs [50–53], while others incorporate encryption-based mechanisms [54,55] that ensure sensitive data can only be decrypted by authorised parties, even if $\mathcal{A}_{middle}$ has internal access. However, a few protocols [56–58] are unable to fully resist internal threats, as in certain scenarios, even relatively weak accomplices may gain access to sensitive information.

Considering $\mathcal{A}_{strong}$, the majority of protocols can resist attacks, as they incorporate data obfuscation or decentralised control mechanisms, making it difficult even for $\mathcal{A}_{strong}$ to access sensitive information [50,53,59–61]. However, some protocols remain vulnerable [62–64], particularly when internal adversaries are able to access delegation tokens or manipulate session data.

6.3.2. Integrity

Evaluating integrity protection, it was found that all protocols are secure against $\mathcal{A}_{weak}$, as this adversary lacks corruption capabilities and is, therefore, unable to tamper with the data.

$\mathcal{A}_{middle}$ faces significant challenges in tampering with data integrity in most of the considered protocols, as these protocols are designed with strong encryption and authentication mechanisms. Many of them are based on blockchain technology, smart contracts, and secure encryption schemes to protect data from unauthorised modification [60,65–67]. However, some protocols may still be vulnerable to $\mathcal{A}_{middle}$'s attacks if internal accomplices are able to tamper with access tokens and thereby influence the delegation process [63,64,68]. The same holds true for $\mathcal{A}_{strong}$. Although this adversary has more advanced capabilities than $\mathcal{A}_{middle}$, they cannot compromise the integrity of systems based on blockchain, as any modification must pass a network-wide validation process. To manipulate the entire network, $\mathcal{A}_{strong}$ would need to control more than 50% of it, which is considered an unlikely scenario.

6.3.3. Availability

Examining the availability property, it is observed that the percentage of systems protected against $\mathcal{A}_{weak}$ in this domain is lower than in the previous two. Most protocols are vulnerable to $\mathcal{A}_{weak}$'s attacks, e.g., DoS, except for a few that employ techniques such as rate limiting, traffic filtering, load balancing, or redundancy mechanisms [23,56,61].

Furthermore, most protocols are also not protected against $\mathcal{A}_{middle}$ and $\mathcal{A}_{strong}$, as both adversaries have the capability to manipulate internal resources, bypass external protections, and disrupt services. Even if the entire system is not compromised, in certain application areas, e.g., healthcare, the availability disruption of even a single device can be critical [56,67,69,70].

6.4. Computation Delegation

In this domain, existing approaches focus on delegating computational tasks that are infeasible or resource-intensive for RC systems to perform independently. These solutions pay considerable attention to data protection during storage and transmission, often employing a combination of techniques such as digital signatures, encryption, and access control mechanisms to restrict data exposure, even among legitimate system participants.

However, some approaches prioritise the organisational aspects of delegation and provide limited details on the specific security technologies employed.

The overall architecture of protocols related to this domain is illustrated in Figure 2d, and the summarised results of their security evaluation are presented in Figure 7d.

6.4.1. Confidentiality

The majority of the protocols in this domain provide confidentiality through security techniques such as encryption, key management, and secure communication protocols, e.g., DTLS. These techniques ensure that even if $\mathcal{A}_{weak}$ tries to intercept the communication, the data remain inaccessible without the appropriate keys or permissions [15,71,72]. However, we have labelled some protocols as not protected against $\mathcal{A}_{weak}$, particularly those that do not mention the use of cryptographic mechanisms, as they are primarily focused on data processing and analytics [73–75]. These protocols might not focus on ensuring confidentiality in terms of encryption or secure data transmission and may be vulnerable to data exposure to $\mathcal{A}_{weak}$ because they do not incorporate sufficient protection mechanisms. The same holds true for $\mathcal{A}_{middle}$. Having accomplices within the system may provide an advantage for $\mathcal{A}_{middle}$ over $\mathcal{A}_{weak}$ only in the case of a few protocols [15,76].

Considering attacks against confidentiality from $\mathcal{A}_{strong}$, it is observed that more than half of the protocols remain resistant [20,71,77,78]. These protocols generally employ architectures that prevent internal adversaries from directly accessing or modifying data, even with full system access. However, several protocols are still vulnerable to $\mathcal{A}_{strong}$, particularly those that focus primarily on data-processing or -management frameworks [73,79] and do not incorporate sufficient security mechanisms to protect data against internal threats

6.4.2. Integrity

Most protocols are secure against $\mathcal{A}_{weak}$ in terms of integrity, as $\mathcal{A}_{weak}$ lacks the ability to tamper with data. However, some protocols do not mention any mechanisms for ensuring data integrity and are therefore classified as insecure [13,80–82].

Considering $\mathcal{A}_{midde}$, it is visible that the majority of discussed protocols are able to resist it in terms of integrity. Most of these protocols incorporate integrity verification measures, e.g., message authentication codes, digital signatures, end-to-end encryption, and hashing to detect and prevent unauthorised data tampering [71,78,83]. However, some protocols do not employ such measures and, hence, may be vulnerable to $\mathcal{A}_{midde}$'s attacks, which can be performed with the assistance of $\mathcal{A}_{midde}$'s accomplices [78,84]. Similar conclusions can be applied to $\mathcal{A}_{strong}$; however, as they have more resources than $\mathcal{A}_{midde}$, a higher number of protocols are vulnerable to their attacks [8,20,85,86].

6.4.3. Availability

When evaluating the availability of the systems, it is observed that the percentage of vulnerable systems is relatively low, less than half of the protocols are vulnerable regarding all three types of adversaries.

As an external adversary, $\mathcal{A}_{weak}$ can perform attacks only on parties that accept requests from outside the system and cause a single point of failure or overload the network and disrupt the system's availability [76,87,88].

$\mathcal{A}_{midde}$ and $\mathcal{A}_{strong}$ have also capabilities to disrupt the system functionality from inside; hence, more protocols are vulnerable to their attacks [89–91], as they can bypass traffic filtering mechanisms.

6.5. Common Vulnerabilities

Analysing the attacks to which systems are vulnerable, several key reasons for confidentiality breaches were identified. First, many systems rely on sensors and resource-constrained IoT nodes that store sensitive information but lack sufficient built-in security. As a result, adversaries with the capability to corrupt internal system components (such as $\mathcal{A}_{middle}$ or $\mathcal{A}_{strong}$) can access sensor data, disclose it, or leverage it for future attacks [28,34,39,47]. Second, user or client confidentiality is often compromised through fraudulent schemes in which users are tricked into voluntarily providing their credentials. This allows adversaries to impersonate legitimate users, effectively performing spoofing attacks. Third, insecure communication channels and transport protocols are another major vulnerability, as they are susceptible to eavesdropping, allowing adversaries to intercept sensitive data during transmission [15,40,62,92].

Integrity violation attacks often occur as a consequence of successful spoofing. Once an adversary has deceptively gained access to a user's resources, they can modify or manipulate those resources by leveraging the user's privileges [21,28,37]. Furthermore, many integrity breaches result from the use of insecure communication channels, enabling the adversary to launch MitM attacks. In addition, sensors are frequent targets of tampering due to their limited protection mechanisms. This allows the adversary to falsify sensor readings, which can have critical implications, particularly in healthcare systems, where accurate data are crucial for patients [42,43,56].

The majority of system availability violations are related to DoS attacks, which are often feasible even for the weakest adversary ($\mathcal{A}_{weak}$) [93,94]. For example, attacking sensors can disrupt the availability of their data. In some cases, this may not lead to critical issues, but in domains such as healthcare, data from devices like heart rate or oxygen monitors are crucial and must be available on demand [86]. More resourceful adversaries, with either indirect (through accomplices) or direct access to the system, are capable of targeting more critical components, e.g., databases or servers, via DoS attacks. Often, such attacks can lead to SPoF, causing a complete system outage. Additionally, powerful adversaries ($\mathcal{A}_{strong}$) that are insiders may be able to restrict legitimate user access depending on their position within the system, thereby compromising availability [85,87,95].

Summarising this section, it can be concluded that systems across all application domains are susceptible to similar types of attacks. The most common issues related to breaches of confidentiality and integrity are caused by the disclosure of credentials by users or clients, often as a result of social engineering or fraudulent schemes. Regarding system availability, the most frequent cause of disruption is a DoS attack targeting critical system components that lack redundancy, resulting in SPoF.

6.6. Proposed Countermeasures and Mitigations

This section is dedicated to the discussion of the countermeasures proposed by the authors of the reviewed papers, which aim to prevent various attacks and ensure the fundamental security properties (CIA).

In order to ensure the confidentiality and integrity of the information, several systems implement the TLS protocol directly [28,30,33,96–98] or use it as a basis for the DTLS handshake protocol [21,22,38,42,47,48,85,99–102]. DTLS not only secures transmitted data and provides strong authentication inherited from TLS, but also helps accelerate system processes. To counter long-term attacks, dynamic monitoring is proposed [103], where the system observes device activity during runtime. Based on this observation, it can detect suspicious behaviour and impose restrictions if necessary. Additionally, various combinations of lightweight cryptographic protocols have been suggested to ensure data protection, each offering specific advantages and trade-offs depending on the application

context [8,15,94,104]. Furthermore, many systems adopt security mechanisms such as cryptographic hashing and multi-factor authentication to enhance data protection. Decentralised architectures, including blockchain-based solutions, are also widely implemented to ensure tamper resistance, transparency, and trust [22,46,49,61,105,106].

According to the results of our analysis, DoS attacks that lead to SPoF pose the greatest risk to system availability. Delegation, by design, helps reduce the success rate of DoS attacks by shifting the adversary's target from a resource-constrained device to a more capable delegate. Since a delegate is assumed to have greater computational and communication resources, it is generally expected to withstand such attacks. However, in practice, a delegate may be more powerful than other devices in the system, but still not strong enough to resist a determined adversary. If an adversary succeeds in corrupting a delegate, this can result in a SPoF scenario that compromises the availability of the entire system. In order to mitigate risks associated with a SPoF, several countermeasures have been implemented in the evaluated systems. These include accepting requests only from already authenticated users [23,34,38,43,45,47,97,101,102], deploying multiple proxies instead of a single delegate to distribute the load and reduce reliance on a single point [20], and implementing decentralised or redundant architectures, often based on blockchain technology [8,32,33,37,45,46,61,107]. These strategies significantly reduce or even eliminate the impact of successful DoS attacks, thereby enhancing system availability. However, despite their undeniable advantages, for example, blockchain-based approaches are not always applicable for resource-constrained devices, which prevents them from being used for typical systems such as smart home, which mostly consists of "weak" devices.

Summarising the results of our research and discussion, the following conclusions are drawn:

- Most existing studies do not provide formalised models of their proposed solutions, with the notable exception of those within the cryptographic delegation application domain.
- The security analysis of the proposed delegation solutions is often either completely absent or limited to a narrow scope, typically considering only a single type of adversary. In most cases, the evaluated adversary is weak and external, while stronger internal threats or more complex attack scenarios are largely overlooked.
- Most of the proposed solutions meet confidentiality and integrity requirements when evaluated against specific types of adversaries. However, they often do not provide sufficient system availability, especially under DoS attacks or in the presence of a potential SPoF.
- Solutions that provide robust and comprehensive security guarantees are frequently tailored for environments with ample computational and network resources. Consequently, their applicability to typical IoT scenarios, e.g., smart home, may be limited, as these scenarios often lack the necessary infrastructure, device density, or processing capabilities required to support such solutions effectively.

7. Potential Future Research Directions

Based on the results of our research, the following potential future research directions in the domain of IoT security delegation may be considered particularly interesting:

Comprehensive and lightweight delegation solutions: Investigate the feasibility of designing delegation-based security frameworks that provide comprehensive protection of confidentiality, integrity, and availability, even in the presence of strong and internal adversaries, while remaining suitable for deployment in small-scale and highly resource-constrained IoT environments.

Formalisation of attacks: Explore whether common attack scenarios targeting delegation mechanisms in IoT systems can be generalised and expressed as formal mathematical models compatible with our security property definitions.
Ensuring security of a delegate: Research techniques to enhance the security of delegates, which are often assumed to be trustworthy and robust components. However, as they frequently serve as main components in delegation models, they become attractive targets for adversaries and may be vulnerable to various attacks.
Delegation in dynamic environments: Design adaptive delegation mechanisms that can respond to real-time contextual factors such as device failure, workload, trust levels, and network conditions, thereby improving both system robustness and efficiency in dynamic IoT environments.
Standardisation and interoperability: Investigate the feasibility of designing interoperable and standardised delegation solutions capable of operating across heterogeneous IoT environments, addressing the current limitations of many existing approaches that are highly domain- or vendor-specific.
Extended property evaluation: Explore the extension of the security evaluation model proposed in this paper to include additional security properties beyond those discussed, in order to support a more comprehensive assessment of system security.

We believe that these research directions provide a solid foundation for future advancements in delegation-based security for IoT systems. Addressing these challenges not only enhances the robustness of existing solutions but also promotes the development of scalable, interoperable, and adaptive frameworks suitable for diverse and resource-constrained IoT environments.

8. Conclusions

In this paper, we performed a comprehensive review of 256 existing solutions that propose delegation-based mechanisms for securing IoT systems. Through this analysis, four main application domains were identified: access control and/or authorisation delegation, authentication delegation, rights delegation, and computation delegation. For each domain, a generalised system architecture was derived based on common patterns observed across the solutions included in the category. Furthermore, a flexible mathematical model of a potential adversary was introduced, along with a formalisation of the confidentiality, integrity, and availability security properties relevant to each domain. These properties were expressed in the form of security games, in which an adversary attempts to win the game by violating the given property. Finally, all considered delegation solutions were evaluated in terms of their robustness against three distinct types of adversaries with respect to the introduced security games.

According to the obtained results, it can be concluded that despite the existence of many delegation approaches ensuring information confidentiality and integrity, the majority remain vulnerable to attacks, which target system availability, particularly those resulting in a single point of failure. Although delegation-based security mechanisms offer clear advantages over non-delegation approaches, especially in supporting resource-constrained IoT devices, there is still considerable room for improvement. Notably, delegation solutions face the dual challenge of securing both the constrained devices and the delegates themselves, which frequently become prime targets for adversaries.

Author Contributions: Conceptualisation, E.G. and S.K.; methodology, E.G. and P.S.; validation, E.G. and F.K.; formal analysis, E.G. and F.K.; investigation, E.G.; data curation, E.G.; writing—original draft preparation, E.G., F.K., P.S. and S.K.; writing—review and editing, E.G., F.K., P.S. and S.K.; visualisation, E.G.; supervision, S.K. All authors have read and agreed to the published version of the manuscript.

Funding: The work was funded by the Bavarian State Ministry of Science and Arts (BayStMWK), under the project "Secure Encapsulation" of the Bavarian Research Association "FORDaySec" (see https://fordaysec.de (accessed on 12 March 2025)) "Security in Everyday Use of Digital Technologies".

Data Availability Statement: No new data were created or analyzed in this study.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IoT	Internet of Things
CIA	Confidentiality, Integrity and Availability
TLS	Transport Layer Security
SoK	Systematisation of Knowledge
ACM	Association for Computing Machinery
IEEE	Institute of Electrical and Electronics Engineers
RC	Resource-constrained
DTLS	Datagram Transport Layer Security
SP	Security Proxy
ID	Identification Number
SA	Security Agent
LW-RHABE	Lightweight Revocable Hierarchical Attribute-based Encryption
CA	Central Authority
DA	Domain Authority
CSP	Cloud Service Provider
DoS	Denial-of-Service
MitM	Man-in-the-Middle
SPoF	Single Point of Failure

Appendix A. Methodology for Selecting Papers Suitable for the Review

In this section, the approach used to collect existing publications on delegated IoT security for this survey is described.

Appendix A.1. Selection Procedure

In order to find suitable publications, the following scientific libraries were used: Google Scholar, Semantic Scholar, ACM Digital Library, and IEEE Xplore. Currently, these platforms provide access to more than 203 million papers across diverse research disciplines. Following Kitchenham et al., a keyword query was formulated to identify scientific works relevant to this study [108]:

$$(\text{“}\textit{Internet of Things}\text{”} \mid\mid \text{“}\textit{IoT}\text{”}) \\ \wedge \\ (\text{“}\textit{Delegated}\text{”} \mid\mid \text{“}\textit{Delegated Security}\text{”} \mid\mid \text{“}\textit{Delegation}\text{”}).$$

Additionally, applying the snowballing strategy introduced by Wohlin and Prikladniki [109], the references of the previously identified papers were examined and searched for relevant publications containing additional security models and designs applicable to our study. After collecting the papers retrieved through this process, the next phase was initiated.

In order to establish a consistent standard among the selected papers, all collected papers were filtered based on the following criteria:

Inclusion criteria:

$(IC1)$ A paper provides a unique approach toward the definition, description, or classification of security delegating in IoT systems.

$(IC2)$ A paper presents metrics or security models that allow for explaining or measuring security delegation.

$(IC3)$ A paper introduces a specific approach to security delegation that can be applied in at least one sub-area of IoT.

Exclusion criteria:

$(EC1)$ A paper is not peer-reviewed.

$(EC2)$ A paper does not present a novel security model or design.

A paper is considered relevant if at least one inclusion criterion is met and none of the exclusion criteria apply:

$$(IC1 \vee IC2 \vee IC3) \wedge \overline{(EC1 \vee EC2)} = 1.$$

Appendix A.2. Results of the Selection Procedure

After completing the search phase of the selection procedure, a list of 940 publications was obtained. The filtering process then began with an initial scan to assess whether each publication might be relevant. At this stage, only papers clearly unrelated to our topic were excluded, for example, those whose subject matter did not fall within the IoT domain. As a result, 404 papers remained.

In the second round, all remaining papers were examined to determine whether they met the inclusion or exclusion criteria (see Appendix A.1). Subsequently, each paper was categorised as either relevant or not relevant. Finally, 254 papers were selected for the review.

Table A1 contains all papers retrieved during the selection procedure which are grouped according to the categories identified in Section 2. The *Rights Delegation* category in this table is split into two parts since papers #153-192 focus on Delegated Proof of Stake, Delegated Proof of Authority, etc. While these papers fall under the rights delegation category, they are primarily concerned with consensus protocols. Therefore, although they do involve the delegation of rights, we have chosen to partially highlight them. Special designations used in the table: **Y**—a year of publication, **T**—a type of the paper (**C**onference, **J**ournal), **Cn**—number of citations, **h** is an *h*-index.

Table A1. List of the papers retrieved during the selection procedure.

№	Title	Y	T	Cn	h
	Access Control and/or Authorisation Delegation				
1	Authorisation framework for the Internet-of-Things [110]	2013	C	70	/
2	A DTLS-based security architecture for the Internet of Things [25]	2015	C	24	/
3	A delegated authorisation solution for smart-city mobile applications [97]	2016	C	4	/
4	Access control framework for API-enabled devices in smart buildings [111]	2016	C	18	/
5	An overview on delegated authorisation for CoAP: Authentication and authorisation for … (ACE) [99]	2016	C	8	/
6	Building secure healthcare services using OAuth 2.0 and JSON web token in IOT cloud scenario [30]	2016	C	23	/
7	IoT Delegate: Smart Home Framework for Heterogeneous IoT Service Collaboration [112]	2016	J	15	34
8	IoT-cloud authorisation and delegation mechanisms for ubiquitous sensing and actuation [34]	2016	C	8	/
9	TACIoT: multidimensional trust-aware access control system for the Internet of Things [113]	2016	J	109	102
10	A Community-Driven Access Control Approach in Distributed IoT Environments [114]	2017	J	55	272
11	A model to enable application-scoped access control as a service for IoT using OAuth 2.0 [29]	2017	C	24	/
12	Achievable Multi-Security Levels for Lightweight IoT-Enabled Devices in …Communications [115]	2017	J	24	204
13	ControlChain: Blockchain as a Central Enabler for Access Control Authorisations in the IoT [31]	2017	C	95	/
14	OAuth-IoT: An access control framework for the Internet of Things based on open standards [28]	2017	C	45	/
15	User-centric access control for efficient security in smart cities [116]	2017	C	11	/
16	A Group-Oriented DTLS Handshake for Secure IoT Applications [21]	2018	J	21	102
17	Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT [26]	2018	J	905	149
18	Decentralised IoT Data Management Using BlockChain and Trusted Execution Environment [32]	2018	C	98	/
19	Group-Oriented Range-Bound Key Agreement for Internet of Things Scenarios [117]	2018	J	18	149
20	Heracles: Scalable, Fine-Grained Access Control for Internet-of-Things in Enterprise Environments [118]	2018	C	14	/
21	Validation of an Adaptive Risk-based Access Control Model for the Internet of Things [103]	2018	J	59	/
22	5G-SSAAC: Slice-specific Authentication and Access Control in 5G [119]	2019	C	5	/
23	Blockchain and Structural Relationship-Based Access Control for IoT: A Smart City Use Case [27]	2019	C	6	/
24	Dynamic Multiparty Authentication using Cryptographic Hardware for the Internet of Things [120]	2019	C	3	/
25	Enabling Decentralised Identifiers and Verifiable Credentials for Constrained …Devices … [121]	2019	J	33	4
26	Fine-grained multi-authority access control in IoT-enabled mHealth [36]	2019	J	24	43
27	Policy-based access control for constrained healthcare resources in the context of the Internet of Things [35]	2019	J	47	129
28	Trusted D2D-Based IoT Resource Access Using Smart Contracts [33]	2019	C	7	/
29	A new scalable authentication and access control mechanism for 5G-based IoT [122]	2020	J	29	151
30	Access control for Internet of Things—enabled assistive technologies: … [37]	2020	J	11	/

Table A1. *Cont.*

№	Title	Y	T	Cn	h
31	BDSS-FA: A Blockchain-Based Data Security Sharing Platform With Fine-Grained Access Control [123]	2020	J	41	204
32	Droplet: Decentralized Authorisation and Access Control for Encrypted Data Streams [17]	2020	J	30	4
33	Reliable Task Management Based on a Smart Contract for Runtime Verification … [124]	2020	J	34	219
34	User-Managed Access Delegation for Blockchain-driven IoT Services [125]	2020	C	4	/
35	Delegated Device Attestation for IoT [96]	2021	C	1	/
36	A Proxy Signature-Based Swarm Drone Authentication With Leader Selection in 5G Networks [126]	2022	J	10	204
37	Perils and Mitigation of Security Risks of Cooperation in Mobile-as-a-Gateway IoT [18]	2022	C	1	/
38	Securing Smart Home IoT Systems with Attribute-Based Access Control [127]	2022	C	5	/
39	A Blockchain-Based Framework for Scalable and Trustless Delegation of Cyber Threat Intelligence [106]	2023	C	1	/
40	An access control scheme for distributed IoT based on adaptive trust evaluation and blockchain [128]	2023	J	2	5
41	An extended Attribute-based access control with controlled delegation in IoT [129]	2023	J	1	54
42	Dynamic Secure Access Control and Data Sharing Through Trusted Delegation and Revocation … [130]	2023	J	5	149
43	Multi-tenant, Decentralized Access Control for the Internet of Things [131]	2023	C	0	/
44	Multilevel Subgranting by Power of Attorney and OAuth Authorisation Server … [132]	2023	J	0	149
45	Smart Blockchain-based Authorisation for Social Internet of Things [12]	2023	C	0	/
46	Flexible and Fine-grained Access Control for EHR in Blockchain-assisted E-healthcare Systems [133]	2024	J	0	149
47	Privacy-Preserving Fine-Grained Data Sharing With Dynamic Service for the Cloud-Edge IoT [134]	2025	C	2	/
	Authentication Delegation				
48	A Simple Delegation Scheme for RFID Systems (SiDeS) [49]	2007	C	14	/
49	End-to-end transport-layer security for Internet-integrated sensing applications with … [47]	2013	J	11	9
50	Towards viable certificate-based authentication for the Internet of Things [41]	2013	C	91	/
51	Delegation-based authentication and authorisation for the IP-based Internet of Things [100]	2014	C	102	/
52	Lightweight secure communication for CoAP-enabled Internet of Things using delegated DTLS … [38]	2014	C	21	/
53	An OAuth based authentication mechanism for IoT networks [39]	2015	C	39	/
54	ESSE: Efficient Secure Session Establishment for Internet-Integrated Wireless Sensor Networks [101]	2015	J	20	65
55	IoT-OAS: An OAuth-Based Authorisation Service Architecture for Secure Services in IoT Scenarios [40]	2015	J	174	145
56	SEA: A Secure and Efficient Authentication and Authorisation Architecture for IoT-Based Healthcare … [42]	2015	J	222	/
57	Toward a Lightweight Authentication and Authorisation Framework for Smart Objects [135]	2015	J	111	251
58	An Authentication and Key Management Mechanism for Resource Constrained Devices … [45]	2017	J	22	219
59	IoT–Cloud collaboration to establish a secure connection for lightweight devices [48]	2017	J	25	98

Table A1. *Cont.*

№	Title	Y	T	Cn	h
60	Secure Service Proxy: A CoAP(s) Intermediary for a Securer and Smarter Web of Things [102]	2017	J	9	219
61	BF-IoT: Securing the IoT Networks via Fingerprinting-Based Device Authentication [43]	2018	C	32	/
62	A DPN (Delegated Proof of Node) Mechanism for Secure Data Transmission in IoT Services [136]	2019	J	20	51
63	D2TLS: delegation-based DTLS for cloud-based IoT services [22]	2019	C	6	/
64	Privacy-preserving delegable authentication in the Internet of Things [16]	2019	C	11	/
65	Master-slave chain based trusted cross-domain authentication mechanism in IoT [46]	2020	J	31	129
66	Cost-Efficient Anonymous Authentication Scheme Based on Set-Membership Zero-Knowledge Proof [14]	2023	C	0	/
67	Leakage of Authorisation-Data in IoT Device Sharing: New Attacks and Countermeasure [44]	2023	J	0	92
68	Zero-Knowledge Proofs based delegation authentication for Industrial Internet of Things … [19]	2023	C	0	/
	Rights Delegation				
69	Capability-based access control delegation model on the federated IoT network [93]	2012	J	9	9
70	A Hot-topic based Distribution and Notification of Events in Pub/Sub Mobile Brokers [137]	2013	J	8	/
71	A capability-based security approach to manage access control in the Internet of Things [57]	2013	J	299	116
72	Distributed Capability-based Access Control for the Internet of Things [138]	2013	J	178	9
73	DCapBAC: embedding authorisation logic into smart things through ECC optimisations [68]	2016	J	125	54
74	FairAccess: a new Blockchain?based access control framework for the Internet of Things [50]	2016	J	650	58
75	Dynamic Access Control Policy based on Blockchain and Machine Learning for the Internet of Things [139]	2017	J	173	35
76	Offline Trusted Device and Proxy Architecture Based on a new TLS Switching Technique [92]	2017	C	1	/
77	RSPP: A reliable, searchable and privacy-preserving e-healthcare system … [140]	2017	C	31	/
78	ViotSOC: Controlling Access to Dynamically Virtualized IoT Services using Service Object Capability [141]	2017	C	4	/
79	BlendCAC: A BLockchain-Enabled Decentralized Capability-Based Access Control for IoTs [52]	2018	C	103	/
80	BlendCAC: A Smart Contract Enabled Decentralized Capability-Based Access Control … [51]	2018	J	121	32
81	Blockchain-Based IoT-Cloud Authorisation and Delegation [142]	2018	C	41	/
82	CapChain: A Privacy Preserving Access Control Framework Based on Blockchain … [53]	2018	C	29	/
83	IoTChain: A blockchain security architecture for the Internet of Things [65]	2018	C	146	/
84	Pragmatic approach using OAuth mechanism for IoT device authorisation in cloud [62]	2018	C	2	/
85	A Novel Entitlement-based Blockchain-enabled Security Architecture for IoT [143]	2019	C	17	/
86	A Secure Key Delegation Mechanism for Fog Networking [144]	2019	C	4	/
87	Access Control Model Based on Dynamic Delegations and Privacy in a Health System … [63]	2019	C	0	/
88	Blockchain Based Fine-Grained and Scalable Access Control for IoT Security and Privacy [145]	2019	C	8	/

Table A1. *Cont.*

№	Title	Y	T	Cn	h
89	Blockchain based permission delegation and access control in Internet of Things (BACI) [146]	2019	J	70	112
90	CA-ADP: Context-Aware Authorisation and Delegation Protocol for IoT-based healthcare smart systems [56]	2019	C	2	/
91	Context-aware pseudonymisation and authorisation model for IoT-based smart hospitals [147]	2019	J	11	64
92	DCACI: A Decentralized Lightweight Capability Based Access Control Framework using IOTA … [148]	2019	C	18	/
93	Design and implementation of a secure and flexible access-right delegation for … [58]	2019	J	21	151
94	Tc-PEDCKS: Towards time controlled public key encryption with … keyword search … [149]	2019	J	22	129
95	WAVE: A Decentralized Authorisation Framework with Transitive Delegation [61]	2019	J	68	4
96	A Flexible Privacy-Preserving Data Sharing Scheme in Cloud-Assisted IoT [150]	2020	J	38	149
97	Ciphertext-Policy Hierarchical Attribute-Based Encryption Against Key-Delegation Abuse … [64]	2020	J	9	204
98	Exploiting Smart Contracts for Capability-Based Access Control in the Internet of Things [151]	2020	J	67	219
99	On the Design of a Flexible Delegation Model for the Internet of Things Using Blockchain [23]	2020	J	33	170
100	On the Integration of Blockchain to the Internet of Things for Enabling Access Right Delegation [152]	2020	J	42	149
101	Proxy re-encryption with equality test for secure data sharing in IoT-based healthcare systems [69]	2020	J	11	/
102	Towards Decentralized IoT Updates Delivery Leveraging Blockchain and Zero-Knowledge Proofs [153]	2020	C	6	/
103	xDBAuth: Blockchain Based Cross Domain Authentication and Authorisation Framework for IoT [154]	2020	J	56	204
104	A Secure and Privacy-Preserving Machine Learning Model Sharing Scheme for Edge-Enabled IoT [155]	2021	J	13	204
105	An Efficient Access Control Scheme With Outsourcing and Attribute Revocation … [156]	2021	J	17	204
106	Blockchain-Based DNS Root Zone Management Decentralisation for Internet of Things [157]	2021	J	5	73
107	Cost-Effective Proxy Signcryption Scheme for Internet of Things [158]	2021	J	3	42
108	Entitlement-Based Access Control for Smart Cities Using Blockchain [159]	2021	J	10	219
109	FogFrame: a framework for IoT application execution in the fog [160]	2021	J	15	37
110	On Designing Context-Aware Trust Model and Service Delegation for Social Internet of Things [66]	2021	J	24	149
111	A Bidirectional Trust Model for Service Delegation in Social Internet of Things [161]	2022	J	6	4
112	A Novel MQTT 5.0-Based Over-the-Air Updating Architecture Facilitating Stronger Security [162]	2022	J	10	62
113	A Proxy Re-Encryption Approach to Secure Data Sharing in the IoT Based on Blockchain [60]	2022	J	58	98
114	A Study on Vehicle Monitoring Service Using Attribute-Based Security Scheme … [163]	2022	J	1	101
115	A Traceable Capability-based Access Control for IoT [164]	2022	J	3	51
116	An Efficient Blockchain-Based Hierarchical Data Sharing for Healthcare Internet of Things [67]	2022	J	39	170

Table A1. *Cont.*

№	Title	Y	T	Cn	h
117	An Industrial IoT-Based Blockchain-Enabled Secure Searchable Encryption Approach … [165]	2022	J	95	219
118	Attribute-Based Data Sharing in Smart Healthcare Environment [70]	2022	C	0	/
119	BlueSky: Combining Task Planning and Activity-Centric Access Control for Assistive Humanoid Robots [166]	2022	C	2	/
120	Burn After Reading: Adaptively Secure Puncturable Identity-Based Proxy Re-Encryption Scheme … [167]	2022	J	9	149
121	ConTrust: A Novel Context-Dependent Trust Management Model in Social Internet of Things [168]	2022	J	10	204
122	DSAS: A Secure Data Sharing and Authorized Searchable Framework for e-Healthcare System [169]	2022	J	4	204
123	Delegated Anonymous Credentials With Revocation Capability for IoT …(*DANCIS*) [170]	2022	J	2	149
124	Dynamic Delegation-based Privacy Preserving in IoT Architectures [171]	2022	C	4	/
125	EDTP: Energy and Delay Optimized Trajectory Planning for UAV-IoT Environment [172]	2022	J	18	150
126	Efficient multi-tier, multiple entry PBFT consensus algorithm for IoT [173]	2022	C	3	/
127	How to divide a permission token in the delegation process of blockchain-based access control for IoT [174]	2022	C	0	/
128	Joint Content and Radio Access for the IoT: A Smart-Contract-Based Trusted Framework [59]	2022	J	2	149
129	Monitoring Provenance of Delegated Personal Data with Blockchain [175]	2022	C	2	/
130	Mutual-contained access delegation scheme for the Internet of Things user services [176]	2022	J	29	45
131	Omnes pro uno: Practical Multi-Writer Encrypted Database [177]	2022	J	21	4
132	Redefining the Trust Model for the Internet of Everything in the 6G era [178]	2022	C	1	/
133	Reliable Application Layer Routing Using Decentralized Identifiers [9]	2022	J	1	49
134	SEDIMENT: An IoT-device-centric Methodology for Scalable 5G Network Security [179]	2022	C	0	/
135	Scalable IoT Sensing Systems With Dynamic Sinks [180]	2022	J	3	149
136	Secure Trust-Based Delegated Consensus for Blockchain Frameworks … [181]	2022	J	7	204
137	Secure and Efficient Certificate-Based Proxy Signature Schemes for Industrial Internet of Things [182]	2022	J	11	98
138	Security and Privacy Service Level Agreement composition for Internet of Things systems … [183]	2022	J	2	84
139	Verifiable online/offline multi-keyword search for cloud-assisted Industrial Internet of Things [10]	2022	J	18	54
140	A Fault-Tolerant Distributed Air-to-Ground Communication Architecture for Urban Air Mobility [184]	2023	C	0	/
141	A Linear Homomorphic Proxy Signature Scheme Based on Blockchain for Internet of Things [185]	2023	J	0	64
142	Applying Access Control Enabled Blockchain (ACE-BC) Framework to Manage Data Security … [186]	2023	J	4	219
143	Attribute based access control (ABAC) scheme with a fully flexible delegation mechanism … [54]	2023	J	3	42
144	Blend CAC: Integration for the Blockchain for Distributed Potential Network Access for the IoT [187]	2023	C	1	/
145	Edge-Assisted Intelligent Device Authentication in Cyber–Physical Systems [188]	2023	J	17	149

Table A1. *Cont.*

№	Title	Y	T	Cn	h
146	Key-aggregate searchable encryption with multi-user authorisation and keyword untraceability … [189]	2023	J	0	/
147	SAMP-RPL: secure and adaptive multipath RPL for enhanced security and reliability … [190]	2023	J	7	64
148	SEEMQTT: Secure End-to-End MQTT-Based Communication for Mobile IoT Systems … [191]	2023	J	6	149
149	Blockchain-Enabled Key Aggregate Searchable Encryption Scheme … [105]	2024	J	0	149
150	Identity-Based Multiproxy Signature With Proxy Signing Key for Internet of Drones [77]	2024	J	0	149
151	Multi-objective cost-aware bag-of-tasks scheduling optimisation model for IoT applications … [192]	2024	J	0	150
152	Pairing-Free Certificate-Based Proxy Re-Encryption Plus Scheme for Secure Cloud Data Sharing [55]	2024	J	60	62
153	Roll-DPoS: A Randomized Delegated Proof of Stake Scheme … [193]	2018	C	40	/
154	HyBloSE: hybrid blockchain for secure-by-design smart environments [194]	2020	C	3	/
155	Enabling machine learning-based side-chaining for improving QoS in blockchain-powered IoT networks [195]	2021	J	2	/
156	A Blockchain-Based Privacy Information Security Sharing Scheme in Industrial Internet of Things [196]	2022	J	9	219
157	A Solution for Bilayer Energy-Trading Management in Microgrids Using Multiblockchain [197]	2022	J	10	149
158	A blockchain- and artificial intelligence-enabled smart IoT framework for sustainable city [198]	2022	J	68	101
159	Blockchain-Enhanced Federated Learning Market With Social Internet of Things [199]	2022	J	7	251
160	Blockchain-Governed Federated Transfer Learning for Secure Internet of Drones Networks [200]	2022	J	0	/
161	Cooperative Communication Method Based on Block Chain … [201]	2022	J	3	204
162	Delegated Proof of Accessibility (DPoAC): A Novel Consensus Protocol for Blockchain Systems [202]	2022	J	12	55
163	Delegated Proof of Secret Sharing: A Privacy-Preserving Consensus Protocol … [203]	2022	J	5	/
164	Delegated Proof of Stake Consensus with Mobile Voters and Multiple Entry PBFT Voting [204]	2022	C	1	/
165	Design of a Blockchain-based Security Algorithm for IoT in Healthcare [11]	2022	C	1	/
166	EdgeShare: A blockchain-based edge data-sharing framework for Industrial Internet of Things [205]	2022	J	15	177
167	Investigating Distance Bounding for Delegated Proof-of-Proximity Consensus within IIoT [206]	2022	C	0	/
168	Leveraging Blockchain for Multi-Operator Access Sharing Management in Internet of Vehicles [207]	2022	J	9	204
169	Preferential Delegated Proof of Stake (PDPoS)—Modified DPoS with Two Layers … [208]	2022	J	14	76
170	Secure and Efficient Data Sharing Among Vehicles Based on Consortium Blockchain [209]	2022	J	61	182
171	Sustainable Smart Industry: A Secure and Energy Efficient Consensus Mechanism … [210]	2022	J	18	70
172	A Secure and Intelligent Data Sharing Scheme for UAV-Assisted Disaster Rescue [211]	2023	J	15	179

Table A1. *Cont.*

№	Title	Y	T	Cn	h
173	Achieving a Decentralized and Secure Cab Sharing System Using Blockchain Technology [212]	2023	J	19	182
174	Authentication Technology in IoT and Privacy Security Issues in Typical Application Scenarios [213]	2023	J	4	62
175	Blockchain-Enabled Lightweight Fine-Grained Searchable Knowledge Sharing for Intelligent IoT [214]	2023	J	1	149
176	Efficient Internet-of-Things Cyberattack Depletion … [215]	2023	J	33	219
177	Enabling a Secure IoT Environment Using a Blockchain-Based Local-Global Consensus Manager [216]	2023	J	2	62
178	Improving Quality of Service for Users of Leaderless DAG-Based Distributed Ledgers [217]	2023	J	1	/
179	LightPoW: A trust based time-constrained PoW for blockchain in Internet of Things [218]	2023	J	5	150
180	Optimisation of a Consensus Protocol in Blockchain-IoT Convergence [219]	2023	C	0	/
181	pDPoSt+sPBFT: A High Performance Blockchain-Assisted Parallel Reinforcement Learning … [220]	2023	J	1	65
182	Safe and Efficient Delegated Proof of Stake Consensus Mechanism … [221]	2023	J	1	25
183	Secure Spectrum Sharing for Satellite Internet-of-Things Based on Blockchain [222]	2023	J	0	75
184	A novel blockchain-based digital forensics framework for preserving evidence … [223]	2024	J	0	81
185	Blockchain-enabled trust management for secure content caching in mobile edge computing … [107]	2024	J	0	39
186	Computing Power Networking Meets Blockchain: A Reputation-Enhanced Trading Framework … [224]	2024	J	0	149
187	Enhancing IoT Data Security with Lightweight Blockchain and … [225]	2024	J	0	64
188	On-Chain and Off-Chain Data Management for Blockchain-Internet of Things: A Multi-Agent … [226]	2024	J	0	58
1890	Optimized blockchain-based healthcare framework … [227]	2024	J	2	129
190	Resource Efficient Federated Learning and DAG Blockchain With Sharding … [228]	2024	J	0	149
191	LT-DBFT: A Hierarchical Blockchain Consensus Using Location and Trust in IoT [229]	2025	C	0	/
	Computations Delegation				
192	HIP Tiny Exchange (TEX): A distributed key exchange scheme for HIP-based Internet of Things [91]	2012	C	9	/
193	A federated architecture approach for Internet of Things security [230]	2014	C	83	/
194	C-CP-ABE: Cooperative Ciphertext Policy Attribute-Based Encryption for the Internet of Things [71]	2014	C	46	/
195	Lightweight DTLS Implementation in CoAP-based IoT [72]	2014	J	26	9
196	Lightweight collaborative key establishment scheme for the Internet of Things [8]	2014	J	79	150
197	Mitigating loT Security Threats with a Trusted Network Element [98]	2014	C	19	/
198	Efficient Key Establishment for Constrained IoT Devices with Collaborative HIP-Based Approach [231]	2015	C	5	/
199	Proxy-based end-to-end key establishment protocol for the Internet of Things [20]	2015	C	18	/
200	S3K: Scalable Security With Symmetric Keys—DTLS Key Establishment for the Internet of Things [85]	2016	J	80	102
201	ScriptIoT: A Script Framework for and Internet-of-Things Applications [73]	2016	J	13	149
202	An Ontology-Based Approach for IoT Data Processing Using Semantic Rules [81]	2017	C	6	/
203	CHIP: Collaborative Host Identity Protocol with Efficient Key Establishment … [90]	2017	J	10	75
204	Fast and Parallel Keyword Search Over Public-Key Ciphertexts for Cloud-Assisted IoT [232]	2017	J	11	204

Table A1. *Cont.*

№	Title	Y	T	Cn	h
205	DECENT: Secure and fine-grained data access control with policy updating for constrained IoT devices [233]	2018	J	31	54
206	Efficient Outsourced Data Access Control with User Revocation for Cloud-Based IoT [234]	2018	C	3	/
207	Secure and trusted telemedicine in Internet of Things IoT [86]	2018	C	9	/
208	Securely outsourcing the ciphertext-policy attribute-based encryption [235]	2018	J	21	54
209	Event driven and semantic based approach for data processing on IoT gateway devices [74]	2019	J	17	64
210	IoT meets distributed AI—Deployment scenarios of Bonseyes AI applications on FIWARE [236]	2019	C	6	/
211	JEDI: Many-to-Many End-to-End Encryption and Key Delegation for IoT [87]	2019	J	90	4
212	On-Demand Computation Offloading Architecture in Fog Networks [89]	2019	J	10	62
213	Powering Smart Homes with Information-Centric Networking [80]	2019	J	17	272
214	A Layer-Partitioning Approach for Faster Execution of Neural Network-Based . . . Applications . . . [79]	2020	J	5	204
215	A Lightweight Privacy-Preserving Communication Protocol for Heterogeneous IoT Environment [104]	2020	J	36	204
216	Identity-based encryption with authorized equivalence test for cloud-assisted IoT [237]	2020	J	17	63
217	Lightweight Revocable Hierarchical Attribute-Based Encryption for Internet of Things [15]	2020	J	33	204
218	Lightweighted Secure Searching Over Public-Key Ciphertexts for . . . Industrial IoT Devices [238]	2020	J	27	170
219	Match in My Way: Fine-Grained Bilateral Access Control for Secure Cloud-Fog Computing [239]	2020	J	43	92
220	Privacy-Preserving Computation Offloading for Time-Series Activities Classification in eHealthcare [240]	2020	C	9	/
221	Reliable and secure data transfer in IoT networks [241]	2020	J	20	98
222	Secure Data Sharing and Search for Cloud-Edge-Collaborative Storage [242]	2020	J	33	204
223	A Hybrid Artificial Neural Network for Task Offloading in Mobile Edge Computing [75]	2022	C	5	/
224	A Queueing Game Based Management Framework for Fog Computing . . . [243]	2022	J	22	151
225	Achieving reliable and anti-collusive outsourcing computation and verification . . . [244]	2022	J	5	40
223	An IoT Digital Twin for Cyber-Security Defence Based on Runtime Verification [245]	2022	C	1	/
227	Architecture Blueprints to Enable Scalable Vertical Integration of Assets with Digital Twins [84]	2022	C	3	/
228	Attribute-based searchable encryption with delegated equality test in cloud-assisted Internet of Things [78]	2022	C	0	/
229	Certificateless Group to Many Broadcast Proxy Reencryptions for Data Sharing . . . [246]	2022	J	1	73
230	Edge-assisted Puncturable Fine-grained Task Distribution for the IoT-oriented Crowdsensing [247]	2022	C	0	/
231	Efficient Privacy-Preserving Outsourced Discrete Wavelet Transform in the Encrypted Domain [248]	2022	J	1	61
232	Enabling secure mutual authentication and storage checking in cloud-assisted IoT [249]	2022	J	4	53

Table A1. *Cont.*

№	Title	Y	T	Cn	h
233	Privacy-Preserving and Verifiable Outsourcing Message Transmission and Authentication Protocol … [95]	2022	C	1	/
234	Privacy-preserving CNN feature extraction and retrieval over medical images [250]	2022	J	0	/
235	SecDT: Privacy-Preserving Outsourced Decision Tree Classification … [251]	2022	J	0	36
236	Secure Infectious Diseases Detection System With IoT-Based e-Health Platforms [82]	2022	J	5	149
237	Secure and Temporary Access Delegation With Equality Test for Cloud-Assisted IoV [252]	2022	J	3	182
238	Supporting AI Engineering on the IoT Edge through Model-Driven TinyML [13]	2022	C	4	/
239	A Fog-Based Architecture for Latency-Sensitive Monitoring Applications … [253]	2023	J	2	149
240	A Novel Multi-Party Authentication Scheme for FCN-based MIoT Systems … [254]	2023	J	1	20
241	A key-insulated secure multi-server authenticated key agreement protocol … [255]	2023	J	4	39
242	Computation Offloading for Industrial Internet of Things: A Cooperative Approach [256]	2023	C	0	/
243	Everything Under Control: Secure Data Sharing Mechanism for Cloud-Edge Computing [257]	2023	J	2	154
244	Improving efficiency and security of IIoT communications … [258]	2023	J	4	117
245	LNGate^2: Secure Bidirectional IoT Micro-Payments … [259]	2023	J	0	151
246	Novel proxy signature from lattice for the post-quantum Internet of Things [260]	2023	J	13	64
247	Outsourcing the Computation of Plaintext Encryption for Homomorphic Encryption [88]	2023	C	0	/
248	Management of IoT Devices Data Security Using Blockchain and Proxy Re-encryption Algorithm [83]	2023	C	1	/
249	Proxy-Based Re-Encryption Design for the IoT Ecosystem [261]	2023	C	0	/
250	Towards Fine-Grained Task Allocation With Bilateral Access Control … [262]	2023	J	0	149
251	A lightweight attribute-based signcryption scheme based on cloud-fog assisted in smart healthcare [263]	2024	J	0	404
252	Fog-Assisted Dynamic IoT Device Access Management Using Attribute-Based Encryption [264]	2024	C	0	/
253	QB-IMD: A Secure Medical Data Processing System With Privacy Protection … [265]	2024	J	7	149
254	REEDS: An Efficient Revocable End-to-End Encrypted Message Distribution System for IoT [76]	2024	J	0	92

Appendix B. Results of the Security Evaluation

Table A2 presents a summary of the security evaluation of existing delegation solutions, focusing on confidentiality, integrity, and availability against three types of adversaries. The evaluation was conducted using the security models introduced in Sections 3 and 4. If we believe that a system can be proven secure against a given adversary type, the attack is unlikely or has no critical impact on system functionality, we mark it as (✓). If an attack against the system is feasible and causes significant damage, we mark it as (✗).

Table A2. Results of the Security Evaluation of Existing Delegation Protocols against Three Adversary Types within the Framework of the CIA-Triad (**C**onfidentiality, **I**ntegrity and **A**vailability).

№	Paper Title	C			I			A		
		$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$
Access Control and/or Authorisation Delegation										
1	Seitz et al. [110] (2013)	✓	✗	✗	✓	✗	✗	✗	✗	✗
2	Lessa dos Santos et al. [25] (2015)	✓	✗	✗	✓	✗	✗	✗	✗	✗
3	Sciarretta et al. [97] (2016)	✓	✓	✗	✓	✓	✗	✓	✗	✗
4	Bandara et al. [111] (2016)	✓	✗	✗	✓	✗	✗	✗	✗	✗
5	Beltran and Skarmeta [99] (2016)	✓	✗	✗	✓	✗	✗	✗	✗	✗
6	Solapurkar [30] (2016)	✓	✗	✗	✓	✗	✗	✗	✗	✗
7	Kum et al. [112] (2016)	✓	✗	✗	✓	✗	✗	✗	✗	✗
8	Bruneo et al. [34] (2016)	✓	✗	✗	✓	✓	✗	✓	✗	✗
9	Bernal Bernabe et al. [113] (2016)	✓	✗	✗	✓	✗	✗	✗	✗	✗
10	Hussein et al. [114] (2017)	✓	✗	✗	✓	✗	✗	✗	✗	✗
11	Fernández et al. [29] (2017)	✓	✗	✗	✓	✗	✗	✗	✗	✗
12	Dao et al. [115] (2017)	✓	✗	✗	✓	✗	✗	✗	✗	✗
13	Pinno et al. [31] (2017)	✓	✓	✓	✓	✓	✓	✓	✓	✓
14	Sciancalepore et al. [28] (2017)	✓	✗	✗	✓	✗	✗	✗	✗	✗
15	Beltran et al. [116] (2017)	✓	✗	✗	✓	✗	✗	✗	✗	✗
16	Park and Park [21] (2018)	✓	✓	✗	✓	✗	✗	✗	✗	✗
17	Novo [26] (2018)	✓	✓	✓	✓	✓	✓	✓	✓	✓
18	Ayoade et al. [32] (2018)	✓	✓	✓	✓	✓	✓	✓	✓	✓
19	Chien [117] (2018)	✓	✗	✗	✓	✗	✗	✗	✗	✗
20	Zhou et al. [118] (2018)	✓	✗	✗	✓	✗	✗	✗	✗	✗
21	Atlam et al. [103] (2018)	✓	✗	✗	✓	✓	✓	✓	✗	✗
22	Behrad et al. [119] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗
23	Sabrina [27] (2019)	✓	✓	✓	✓	✓	✓	✓	✓	✓
24	Al-Aqrabi et al. [120] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗
25	Lagutin et al. [121] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗
26	Li et al. [36] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗
27	Pal et al. [35] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗
28	Siris et al. [33] (2019)	✓	✓	✓	✓	✓	✓	✓	✓	✓
29	Behrad et al. [122] (2020)	✓	✗	✗	✓	✗	✗	✗	✗	✗
30	Pal et al. [37] (2020)	✓	✓	✗	✓	✗	✗	✗	✗	✗
31	Xu et al. [123] (2020)	✓	✗	✓	✓	✓	✓	✓	✓	✓
32	Shafagh et al. [17] (2020)	✓	✓	✗	✓	✓	✓	✗	✗	✗
33	Hang and Kim [124] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
34	Lin and Liao [125] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
35	Julku et al. [96] (2021)	✓	✗	✗	✓	✗	✗	✗	✗	✗
36	Abdel-Malek et al. [126] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✗
37	Zhou et al. [18] (2022)	✓	✗	✗	✓	✗	✗	✗	✗	✗
38	Goyal et al. [127] (2022)	✓	✗	✗	✓	✗	✗	✗	✗	✗
39	Dunnett et al. [106] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
40	Jiang et al. [128] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
41	Tegane et al. [129] (2023)	✓	✗	✗	✓	✓	✗	✗	✗	✗
42	Alshehri et al. [130] (2023)	✓	✗	✗	✓	✗	✗	✗	✗	✗
43	Pittaras and Polyzos [131] (2023)	✓	✗	✗	✓	✗	✗	✗	✗	✗
44	Vattaparambil Sudarsan et al. [132] (2023)	✓	✗	✗	✓	✗	✗	✗	✗	✗
45	Dallel et al. [12] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
46	Chen et al. [133] (2024)	✓	✓	✓	✓	✓	✓	✓	✓	✓
47	Sun et al. [134] (2025)	✓	✗	✗	✓	✗	✗	✓	✗	✗

Table A2. *Cont.*

№	Paper Title	C			I			A		
		$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$
Authentication Delegation										
48	Fouladgar and Afifi [49] (2007)	✓	✓	✗	✓	✓	✗	✗	✗	✗
49	Granjal et al. [47] (2013)	✓	✓	✗	✓	✓	✗	✗	✗	✗
50	Hummen et al. [41] (2013)	✓	✓	✗	✓	✓	✗	✗	✗	✗
51	Hummen et al. [100] (2014)	✓	✓	✗	✓	✓	✗	✗	✗	✗
52	Park and Kang [38] (2014)	✓	✓	✗	✓	✓	✗	✓	✓	✗
53	Emerson et al. [39] (2015)	✓	✗	✗	✓	✗	✗	✗	✗	✗
54	Kang et al. [101] (2015)	✓	✓	✗	✓	✓	✗	✓	✓	✗
55	Cirani et al. [40] (2015)	✓	✗	✗	✓	✗	✗	✗	✗	✗
56	Moosavi et al. [42] (2015)	✓	✗	✗	✓	✗	✗	✗	✗	✗
57	Hernandez-Ramos et al. [135] (2015)	✓	✓	✗	✓	✓	✗	✗	✗	✗
58	Kim et al. [45] (2017)	✓	✓	✓	✓	✓	✓	✓	✓	✗
59	Park et al. [48] (2017)	✓	✓	✗	✓	✓	✗	✓	✗	✗
60	Van den Abeele et al. [102] (2017)	✓	✓	✗	✓	✓	✗	✓	✗	✗
61	Gu and Mohapatra [43] (2018)	✓	✗	✗	✓	✗	✗	✓	✗	✗
62	Kim et al. [136] (2019)	✓	✓	✗	✓	✓	✗	✗	✗	✗
63	Cho et al. [22] (2019)	✓	✓	✗	✓	✓	✗	✗	✗	✗
64	Gritti et al. [16] (2019)	✓	✓	✗	✓	✓	✗	✗	✗	✗
65	Guo et al. [46] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
66	Wiraatmaja and Kasahara [14] (2023)	✓	✓	✗	✓	✓	✗	✗	✗	✗
67	Yuan et al. [44] (2023)	✓	✓	✗	✓	✓	✗	✓	✓	✗
68	Rafiqullah et al. [19] (2023)	✓	✓	✗	✓	✓	✗	✗	✗	✗
Rights Delegation										
69	Anggorojati et al. [93] (2012)	✓	✗	✗	✓	✗	✗	✗	✗	✗
70	Morales et al. [137] (2013)	✓	✓	✗	✓	✓	✗	✗	✗	✗
71	Gusmeroli et al. [57] (2013)	✓	✗	✗	✓	✗	✗	✗	✗	✗
72	Hernández-Ramos et al. [138] (2013)	✓	✗	✗	✓	✗	✗	✗	✗	✗
73	Hernández-Ramos et al. [68] (2016)	✓	✗	✗	✓	✗	✗	✗	✗	✗
74	Ouaddah et al. [50] (2016)	✓	✓	✓	✓	✓	✓	✗	✗	✗
75	Outchakoucht et al. [139] (2017)	✓	✓	✓	✓	✓	✓	✗	✗	✗
76	Denis et al. [92] (2017)	✓	✓	✗	✓	✓	✗	✗	✗	✗
77	Yang et al. [140] (2017)	✓	✓	✓	✓	✓	✓	✗	✗	✗
78	Ko et al. [141] (2017)	✓	✓	✗	✓	✓	✗	✗	✗	✗
79	Xu et al. [52] (2018)	✓	✓	✓	✓	✓	✓	✗	✗	✗
80	Xu et al. [51] (2018)	✓	✓	✓	✓	✓	✓	✗	✗	✗
81	Tapas et al. [142] (2018)	✓	✓	✓	✓	✓	✓	✗	✗	✗
82	Le and Mutka [53] (2018)	✓	✓	✓	✓	✓	✓	✗	✗	✗
83	Alphand et al. [65] (2018)	✓	✓	✓	✓	✓	✓	✗	✗	✗
84	Chung et al. [62] (2018)	✓	✗	✗	✓	✗	✗	✗	✗	✗
85	Sabrina [143] (2019)	✓	✓	✓	✓	✓	✓	✗	✗	✗
86	Porwal and Mittal [144] (2019)	✓	✓	✓	✓	✓	✓	✗	✗	✗
87	Mendy et al. [63] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗
88	Sun et al. [145] (2019)	✓	✓	✓	✓	✓	✓	✗	✗	✗
89	Ali et al. [146] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗
90	Zemmoudj et al. [56] (2019)	✓	✗	✗	✓	✓	✗	✓	✗	✗
91	Zemmoudj et al. [147] (2019)	✓	✓	✓	✓	✓	✓	✗	✗	✗
92	Pinjala and Sivalingam [148] (2019)	✓	✓	✓	✓	✓	✓	✗	✗	✗
93	Rabehaja et al. [58] (2019)	✓	✗	✗	✓	✗	✗	✗	✗	✗

Table A2. *Cont.*

№	Paper Title	C			I			A		
		$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$
94	Xu et al. [149] (2019)	✓	✓	✓	✓	✓	✓	✗	✗	✗
95	Andersen et al. [61] (2019)	✓	✓	✓	✓	✓	✓	✓	✗	✗
96	Deng et al. [150] (2020)	✓	✓	✓	✓	✓	✓	✗	✗	✗
97	Chen et al. [64] (2020)	✓	✗	✗	✓	✗	✗	✗	✗	✗
98	Nakamura et al. [151] (2020)	✓	✓	✓	✓	✓	✓	✗	✗	✗
99	Pal et al. [23] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
100	Pal et al. [152] (2020)	✓	✓	✓	✓	✓	✓	✗	✗	✗
101	Li et al. [69] (2020)	✓	✓	✓	✓	✓	✓	✗	✗	✗
102	Puggioni et al. [153] (2020)	✓	✓	✓	✓	✓	✓	✗	✗	✗
103	Ali et al. [154] (2020)	✓	✓	✓	✓	✓	✓	✗	✗	✗
104	Zhou et al. [155] (2021)	✓	✓	✓	✓	✓	✓	✗	✗	✗
105	Zhao et al. [156] (2021)	✓	✓	✓	✓	✓	✓	✗	✗	✗
106	Zhang et al. [157] (2021)	✓	✓	✓	✓	✓	✓	✗	✗	✗
107	Ullah et al. [158] (2021)	✓	✓	✓	✓	✓	✓	✗	✗	✗
108	Sabrina and Jang-Jaccard [159] (2021)	✓	✓	✓	✓	✓	✓	✗	✗	✗
109	Skarlat and Schulte [160] (2021)	✓	✓	✗	✓	✗	✗	✗	✗	✗
110	Wei et al. [66] (2021)	✓	✗	✗	✓	✓	✗	✗	✗	✗
111	Lastname et al. [161] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
112	Chien and Wang [162] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
113	Agyekum et al. [60] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
114	Cha et al. [163] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
115	Li et al. [164] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
116	Zhang et al. [67] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
117	Ali et al. [165] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
118	Kumar and Kumar [70] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
119	Bayreuther et al. [166] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
120	Xiong et al. [167] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
121	Latif [168] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
122	Xue [169] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
123	Pinjala et al. [170] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
124	Silva and Barraca [171] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
125	Banerjee et al. [172] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
126	Qushtom et al. [173] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
127	Heo et al. [174] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
128	Hu et al. [59] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
129	Ju et al. [175] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
130	Panneerselvam and Krithiga [176] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
131	Wang and Chow [177] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
132	Sama et al. [178] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
133	Alsubhi et al. [9] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
134	Shur et al. [179] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
135	Tanyingyong et al. [180] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
136	Goh et al. [181] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
137	Qiao et al. [182] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
138	Rios et al. [183] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
139	Ali et al. [10] (2022)	✓	✓	✓	✓	✓	✓	✗	✗	✗
140	Wanniarachchi and Turau [184] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
141	Wang and Wu [185] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
142	Alharbi [186] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗

Table A2. *Cont.*

№	Paper Title	C			I			A		
		$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$
143	Choksy et al. [54] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
144	Malhotra [187] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
145	Lu et al. [188] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
146	Trivedi and Patel [189] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
147	Sahraoui and Henni [190] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
148	Hamad et al. [191] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
149	Lee et al. [105] (2024)	✓	✓	✓	✓	✓	✓	✗	✗	✗
150	Shin et al. [77] (2024)	✓	✓	✓	✓	✓	✓	✗	✗	✗
151	Seifhosseini et al. [192] (2024)	✓	✓	✓	✓	✓	✓	✗	✗	✗
152	Yan et al. [55] (2024)	✓	✗	✗	✓	✓	✗	✗	✗	✗
153	Fan and Chai [193] (2018)	✓	✗	✗	✓	✗	✗	✗	✗	✗
154	Maselli et al. [194] (2020)	✓	✓	✗	✓	✗	✗	✗	✗	✗
155	Vairagade and Brahmananda [195] (2021)	✓	✓	✗	✓	✗	✗	✗	✗	✗
156	Wang et al. [196] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
157	Huang et al. [197] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
158	Ahmed et al. [198] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
159	Wang et al. [199] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
160	Wang et al. [200] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
161	Zhi et al. [201] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
162	Kaur et al. [202] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
163	Geng et al. [203] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
164	Misic et al. [204] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
165	Rastogi et al. [11] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
166	Yang et al. [205] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
167	Ledwaba et al. [206] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
168	Hu et al. [207] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
169	Bachani and Bhattacharjya [208] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
170	Cui et al. [209] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
171	Sasikumar et al. [210] (2022)	✓	✓	✗	✓	✗	✗	✗	✗	✗
172	Wang et al. [211] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
173	Namasudra and Sharma [212] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
174	Zhao et al. [213] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
175	Wang et al. [214] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
176	Razaque et al. [215] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
177	Alghamdi et al. [216] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
178	Cullen et al. [217] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
179	Qi et al. [218] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
180	Kaur and Gupta [219] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
181	Yang et al. [220] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
182	Mingjie Zhao et al. [221] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
183	Wang et al. [222] (2023)	✓	✓	✗	✓	✗	✗	✗	✗	✗
184	Xiao et al. [223] (2024)	✓	✓	✗	✓	✗	✗	✗	✗	✗
185	Bounaira et al. [107] (2024)	✓	✓	✗	✓	✗	✗	✗	✗	✗
186	Lin et al. [224] (2024)	✓	✓	✗	✓	✗	✗	✗	✗	✗
187	Mohammed and Wahab [225] (2024)	✓	✓	✗	✓	✗	✗	✗	✗	✗
188	Tsang et al. [226] (2024)	✓	✓	✗	✓	✗	✗	✗	✗	✗
189	Al-Marridi et al. [227] (2024)	✓	✓	✗	✓	✗	✗	✗	✗	✗
190	Jiang et al. [228] (2024)	✓	✓	✗	✓	✗	✗	✗	✗	✗
191	Wang et al. [229] (2025)	✓	✗	✗	✓	✗	✗	✓	✗	✗

Table A2. *Cont.*

№	Paper Title	C			I			A		
		$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$
Computations Delegation										
192	Ben Saied and Olivereau [91] (2012)	✓	✓	✗	✓	✗	✗	✓	✗	✗
193	Leo et al. [230] (2014)	✓	✓	✗	✓	✗	✗	✓	✗	✗
194	Touati et al. [71] (2014)	✓	✓	✓	✓	✓	✓	✓	✓	✓
195	Lakkundi and Singh [72] (2014)	✓	✓	✗	✓	✗	✗	✓	✗	✗
196	Saied et al. [8] (2014)	✓	✓	✗	✓	✓	✗	✓	✓	✓
197	Kuusijarvi [98] (2014)	✓	✓	✗	✓	✗	✗	✓	✗	✗
198	Porambage et al. [231] (2015)	✓	✓	✗	✓	✗	✗	✓	✗	✗
199	Porambage et al. [20] (2015)	✓	✓	✗	✓	✓	✗	✓	✓	✗
200	Raza et al. [85] (2016)	✓	✓	✗	✓	✓	✗	✗	✗	✗
201	Hsieh et al. [73] (2016)	✗	✗	✗	✗	✗	✗	✗	✗	✗
202	Bali et al. [81] (2017)	✗	✗	✗	✗	✗	✗	✗	✗	✗
203	Porambage et al. [90] (2017)	✓	✓	✗	✓	✗	✗	✓	✗	✗
204	Xu et al. [232] (2017)	✓	✓	✓	✓	✓	✓	✓	✓	✓
205	Huang et al. [233] (2018)	✓	✓	✓	✓	✓	✓	✓	✓	✓
206	Hao et al. [234] (2018)	✓	✓	✓	✓	✓	✓	✓	✓	✓
207	Albalawi and Joshi [86] (2018)	✓	✓	✗	✓	✓	✗	✗	✗	✗
208	Nguyen et al. [235] (2018)	✓	✓	✓	✓	✓	✓	✓	✓	✓
209	Al-Osta et al. [74] (2019)	✗	✗	✗	✗	✗	✗	✗	✗	✗
210	Moor et al. [236] (2019)	✗	✗	✗	✗	✗	✗	✗	✗	✗
211	Kumar et al. [87] (2019)	✓	✓	✗	✓	✓	✓	✗	✗	✗
212	Jin and Lee [89] (2019)	✓	✓	✗	✓	✗	✗	✓	✗	✗
213	Xu et al. [80] (2019)	✗	✗	✗	✗	✗	✗	✗	✗	✗
214	Saguil and Azim [79] (2020)	✗	✗	✗	✗	✗	✗	✗	✗	✗
215	Luo et al. [104] (2020)	✓	✗	✗	✓	✗	✗	✗	✗	✗
216	Elhabob et al. [237] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
217	Ali et al. [15] (2020)	✓	✗	✗	✓	✓	✓	✗	✗	✗
218	Wang et al. [238] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
219	Xu et al. [239] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
220	Zheng et al. [240] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
221	Gochhayat et al. [241] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
222	Tao et al. [242] (2020)	✓	✓	✓	✓	✓	✓	✓	✓	✓
223	Hamadi et al. [75] (2022)	✗	✗	✗	✗	✗	✗	✗	✗	✗
224	Yi et al. [243] (2022)	✗	✗	✗	✗	✗	✗	✗	✗	✗
225	Wang et al. [244] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
226	De Hoz Diego et al. [245] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
227	Schnicke et al. [84] (2022)	✗	✗	✗	✗	✗	✗	✗	✗	✗
228	Hu et al. [78] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
229	Kim et al. [246] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
230	Jiang and Qin [247] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
231	Zhou et al. [248] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
232	Liu et al. [249] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
233	Li et al. [95] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
234	Cai et al. [250] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
235	Chen et al. [251] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
236	Zhao et al. [82] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
237	Li et al. [252] (2022)	✓	✓	✓	✓	✓	✓	✓	✓	✓
238	Moin et al. [13] (2022)	✗	✗	✗	✗	✗	✗	✗	✗	✗

Table A2. *Cont.*

№	Paper Title	C			I			A		
		$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$	$\mathcal{A}_{weak}$	$\mathcal{A}_{middle}$	$\mathcal{A}_{strong}$
239	Benomar et al. [253] (2023)	✗	✗	✗	✗	✗	✗	✗	✗	✗
240	Meng et al. [254] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
241	Yao et al. [255] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
242	Chouikhi et al. [256] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
243	Song et al. [257] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
244	Atutxa et al. [258] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
245	Kurt et al. [259] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
246	Wang et al. [260] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
247	Li et al. [88] (2023)	✓	✓	✓	✓	✓	✓	✗	✗	✗
248	Mahamuni et al. [83] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
249	Srikanth et al. [261] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
250	Wu et al. [262] (2023)	✓	✓	✓	✓	✓	✓	✓	✓	✓
251	Sun et al. [263] (2024)	✓	✓	✓	✓	✓	✓	✓	✓	✓
252	Routray and Bera [264] (2024)	✓	✓	✓	✓	✓	✓	✓	✓	✓
253	Qu et al. [265] (2024)	✓	✓	✓	✓	✓	✓	✓	✓	✓
254	Li et al. [76] (2024)	✓	✗	✗	✓	✗	✗	✗	✗	✗

References

1. Statista. *Internet of Things—Market Outlook Report*; Statista. Juni 2024. Available online: https://de.statista.com/statistik/studie/id/109209/dokument/internet-der-dinge-market-outlook-report/ (accessed on 28 April 2025).
2. Statista Market Insights. *Annual Number of Cyberattacks Worldwide from 2016 to 2023 (In Millions)*; Graph: 2024. Available online: https://www.statista.com/forecasts/1485031/cyberattacks-annual-worldwide (accessed on 28 April 2025).
3. Miorandi, D.; Sicari, S.; De Pellegrini, F.; Chlamtac, I. Internet of Things: Vision, applications and research challenges. *Ad Hoc Netw.* **2012**, *10*, 1497–1516. [CrossRef]
4. Chacko, A.; Hayajneh, T. Security and Privacy Issues with IoT in Healthcare. *EAI Endorsed Trans. Pervasive Health Technol.* **2018**, *4*, e2. [CrossRef]
5. Canavese, D.; Mannella, L.; Regano, L.; Basile, C. Security at the Edge for Resource-Limited IoT Devices. *Sensors* **2024**, *24*, 590. [CrossRef]
6. Wang, Q.; Li, N.; Chen, H. On the Security of Delegation in Access Control Systems. In Proceedings of the Computer Security—ESORICS 2008, Málaga, Spain, 6–8 October 2008; Jajodia, S., Lopez, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 317–332.
7. Chuat, L.; Abdou, A.; Sasse, R.; Sprenger, C.; Basin, D.; Perrig, A. SoK: Delegation and Revocation, the Missing Links in the Web's Chain of Trust. In Proceedings of the 2020 IEEE European Symposium on Security and Privacy (EuroS&P), Genoa, Italy, 7–11 September 2020; pp. 624–638. [CrossRef]
8. Saied, Y.B.; Olivereau, A.; Zeghlache, D.; Laurent, M. Lightweight collaborative key establishment scheme for the Internet of Things. *Comput. Netw.* **2014**, *64*, 273–295. [CrossRef]
9. Alsubhi, K.; Alzahrani, B.; Fotiou, N.; Albeshri, A.; Alreshoodi, M. Reliable Application Layer Routing Using Decentralized Identifiers. *Future Internet* **2022**, *14*, 322. [CrossRef]
10. Ali, M.; Sadeghi, M.R.; Liu, X.; Miao, Y.; Vasilakos, A.V. Verifiable online/offline multi-keyword search for cloud-assisted Industrial Internet of Things. *J. Inf. Secur. Appl.* **2022**, *65*, 103101. [CrossRef]
11. Rastogi, P.; Singh, D.; Singh Bedi, S. Design of a Blockcdhain based Security Algorithm for IoT in Healthcare. In Proceedings of the 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 28–29 April 2022. [CrossRef]
12. Dallel, O.; Ayed, S.B.; Tahar, J.B.H. Smart Blockchain-based Authorization for Social Internet of Things. In Proceedings of the 2023 International Conference on Cyberworlds (CW), Sousse, Tunisia, 3–5 October 2023; pp. 440–447. [CrossRef]
13. Moin, A.; Challenger, M.; Badii, A.; Gunnemann, S. Supporting AI Engineering on the IoT Edge through Model-Driven TinyML. In Proceedings of the 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), Los Alamitos, CA, USA, 27 June–1 July 2022; pp. 884–893. [CrossRef]

14. Wiraatmaja, C.; Kasahara, S. Cost-Efficient Anonymous Authentication Scheme Based on Set-Membership Zero-Knowledge Proof. In Proceedings of the 2023 5th Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), Paris, France, 11–13 October 2023; pp. 1–8. [CrossRef]
15. Ali, M.; Sadeghi, M.R.; Liu, X. Lightweight Revocable Hierarchical Attribute-Based Encryption for Internet of Things. *IEEE Access* **2020**, *8*, 23951–23964. [CrossRef]
16. Gritti, C.; Önen, M.; Molva, R. Privacy-preserving delegable authentication in the internet of things. In Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, Limassol, Cyprus, 8–12 April 2019; pp. 861–869. [CrossRef]
17. Shafagh, H.; Burkhalter, L.; Ratnasamy, S.; Hithnawi, A. Droplet: Decentralized Authorization and Access Control for Encrypted Data Streams. In Proceedings of the 29th USENIX Security Symposium (USENIX Security 20), Boston, MA, USA, 12–14 August 2020; pp. 2469–2486.
18. Zhou, X.; Guan, J.; Xing, L.; Qian, Z. Perils and Mitigation of Security Risks of Cooperation in Mobile-as-a-Gateway IoT. In Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, Los Angeles, CA, USA, 7–11 November 2022; pp. 3285–3299. [CrossRef]
19. Rafiqullah; Mehmood, A.; Khan, M.A.; Maple, C.; Lloret, J. Zero-Knowledge Proofs based delegation authentication for Industrial Internet of Things in certificateless proxy signatures. In Proceedings of the 2023 10th International Conference on Internet of Things: Systems, Management and Security (IOTSMS), San Antonio, TX, USA, 23–25 October 2023; pp. 8–14. [CrossRef]
20. Porambage, P.; Braeken, A.; Kumar, P.; Gurtov, A.; Ylianttila, M. Proxy-based end-to-end key establishment protocol for the Internet of Things. In Proceedings of the 2015 IEEE International Conference on Communication Workshop (ICCW), London, UK, 8–12 June 2015; pp. 2677–2682. [CrossRef]
21. Park, C.S.; Park, W.S. A Group-Oriented DTLS Handshake for Secure IoT Applications. *IEEE Trans. Autom. Sci. Eng.* **2018**, *15*, 1920–1929. [CrossRef]
22. Cho, E.; Park, M.; Lee, H.; Choi, J.; Kwon, T.T. D2TLS: Delegation-based DTLS for cloud-based IoT services. In Proceedings of the International Conference on Internet of Things Design and Implementation, Montreal, QC, Canada, 15–18 April 2019; pp. 190–201. [CrossRef]
23. Pal, S.; Rabehaja, T.; Hitchens, M.; Varadharajan, V.; Hill, A. On the Design of a Flexible Delegation Model for the Internet of Things Using Blockchain. *IEEE Trans. Ind. Inform.* **2020**, *16*, 3521–3530. [CrossRef]
24. Samonas, S.; Coss, D. The CIA strikes back: Redefining confidentiality, integrity and availability in security. *J. Inf. Syst. Secur.* **2014**, *10*, 21–45.
25. Lessa dos Santos, G.; Guimarães, V.T.; da Cunha Rodrigues, G.; Granville, L.Z.; Tarouco, L.M.R. A DTLS-based security architecture for the Internet of Things. In Proceedings of the 2015 IEEE Symposium on Computers and Communication (ISCC), Larnaca, Cyprus, 6–9 July 2015; pp. 809–815. [CrossRef]
26. Novo, O. Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT. *IEEE Internet Things J.* **2018**, *5*, 1184–1195. [CrossRef]
27. Sabrina, F. Blockchain and Structural Relationship Based Access Control for IoT: A Smart City Use Case. In Proceedings of the 2019 IEEE 44th Conference on Local Computer Networks (LCN), Osnabrück, Germany, 14–17 October 2019; pp. 137–140. [CrossRef]
28. Sciancalepore, S.; Piro, G.; Caldarola, D.; Boggia, G.; Bianchi, G. OAuth-IoT: An access control framework for the Internet of Things based on open standards. In Proceedings of the 2017 IEEE symposium on computers and communications (ISCC), Heraklion, Greece, 3–6 July 2017; pp. 676–681.
29. Fernández, F.; Alonso, A.; Marco, L.; Salvachúa, J. A model to enable application-scoped access control as a service for IoT using OAuth 2.0. In Proceedings of the 2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN), Paris, France, 7–9 March 2017; pp. 322–324. [CrossRef]
30. Solapurkar, P. Building secure healthcare services using OAuth 2.0 and JSON web token in IOT cloud scenario. In Proceedings of the 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), Greater Noida, India, 14–17 December 2016; pp. 99–104. [CrossRef]
31. Pinno, O.J.A.; Gregio, A.R.A.; De Bona, L.C.E. ControlChain: Blockchain as a Central Enabler for Access Control Authorizations in the IoT. In Proceedings of the GLOBECOM 2017—2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6. [CrossRef]
32. Ayoade, G.; Karande, V.; Khan, L.; Hamlen, K. Decentralized IoT Data Management Using BlockChain and Trusted Execution Environment. In Proceedings of the 2018 IEEE International Conference on Information Reuse and Integration (IRI), Salt Lake City, UT, USA, 6–9 July 2018; pp. 15–22. [CrossRef]
33. Siris, V.; Dimopoulos, D.; Fotiou, N.; Voulgaris, S.; Polyzos, G. Trusted D2D-Based IoT Resource Access Using Smart Contracts. In Proceedings of the 2019 IEEE 20th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), Washington, DC, USA, 10–12 June 2019; pp. 1–9. [CrossRef]

34. Bruneo, D.; Distefano, S.; Longo, F.; Merlino, G.; Puliafito, A. IoT-cloud authorization and delegation mechanisms for ubiquitous sensing and actuation. In Proceedings of the 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), Reston, VA, USA, 12–14 December 2016; pp. 222–227. [CrossRef]
35. Pal, S.; Hitchens, M.; Varadharajan, V.; Rabehaja, T. Policy-based access control for constrained healthcare resources in the context of the Internet of Things. *J. Netw. Comput. Appl.* **2019**, *139*, 57–74. [CrossRef]
36. Li, Q.; Zhu, H.; Xiong, J.; Mo, R.; Ying, Z.; Wang, H. Fine-grained multi-authority access control in IoT-enabled mHealth. *Ann. Telecommun.* **2019**, *74*, 389–400. [CrossRef]
37. Pal, S.; Hitchens, M.; Varadharajan, V. Access control for Internet of Things—Enabled assistive technologies: An architecture, challenges and requirements. In *Assistive Technology for the Elderly*; Elsevier: Cambridge, MA, USA, 2020; pp. 1–43. [CrossRef]
38. Park, J.; Kang, N. Lightweight secure communication for CoAP-enabled Internet of Things using delegated DTLS handshake. In Proceedings of the 2014 International Conference on Information and Communication Technology Convergence (ICTC), Busan, Republic of Korea, 22–24 October 2014; pp. 28–33. [CrossRef]
39. Emerson, S.; Choi, Y.K.; Hwang, D.Y.; Kim, K.S.; Kim, K.H. An OAuth based authentication mechanism for IoT networks. In Proceedings of the 2015 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 28–30 October 2015; pp. 1072–1074. [CrossRef]
40. Cirani, S.; Picone, M.; Gonizzi, P.; Veltri, L.; Ferrari, G. IoT-OAS: An OAuth-Based Authorization Service Architecture for Secure Services in IoT Scenarios. *IEEE Sens. J.* **2015**, *15*, 1224–1234. [CrossRef]
41. Hummen, R.; Ziegeldorf, J.H.; Shafagh, H.; Raza, S.; Wehrle, K. Towards viable certificate-based authentication for the internet of things. In Proceedings of the 2nd ACM Workshop on Hot Topics on Wireless Network Security and Privacy, Budapest, Hungary, 19 April 2013; HotWiSec '13, pp. 37–42. [CrossRef]
42. Moosavi, S.R.; Gia, T.N.; Rahmani, A.M.; Nigussie, E.; Virtanen, S.; Isoaho, J.; Tenhunen, H. SEA: A secure and efficient authentication and authorization architecture for IoT-based healthcare using smart gateways. *Procedia Comput. Sci.* **2015**, *52*, 452–459. [CrossRef]
43. Gu, T.; Mohapatra, P. BF-IoT: Securing the IoT Networks via Fingerprinting-Based Device Authentication. In Proceedings of the 2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Chengdu, China, 9–12 October 2018; pp. 254–262. [CrossRef]
44. Yuan, B.; Yang, M.; Xu, Z.; Chen, Q.; Song, Z.; Li, Z.; Zou, D.; Jin, H. Leakage of Authorization-Data in IoT Device Sharing: New Attacks and Countermeasure. *IEEE Trans. Dependable Secur. Comput.* **2023**, *21*, 3196–3210. [CrossRef]
45. Kim, K.; Han, Y.H.; Min, S.G. An Authentication and Key Management Mechanism for Resource Constrained Devices in IEEE 802.11-based IoT Access Networks. *Sensors* **2017**, *17*, 2170. [CrossRef]
46. Guo, S.; Wang, F.; Zhang, N.; Qi, F.; Qiu, X. Master-slave chain based trusted cross-domain authentication mechanism in IoT. *J. Netw. Comput. Appl.* **2020**, *172*, 102812. [CrossRef]
47. Granjal, J.; Monteiro, E.; Silva, J.S. End-to-end transport-layer security for Internet-integrated sensing applications with mutual and delegated ECC public-key authentication. In Proceedings of the 2013 IFIP Networking Conference, Brooklyn, NY, USA, 22–24 May 2013; pp. 1–9.
48. Park, J.; Kwon, H.; Kang, N. IoT—Cloud Collaboration to Establish a Secure Connection for Lightweight Devices. *Wirel. Netw.* **2017**, *23*, 681–692. [CrossRef]
49. Fouladgar, S.; Afifi, H. A Simple Delegation Scheme for RFID Systems (SiDeS). In Proceedings of the 2007 IEEE International Conference on RFID, Grapevine, TX, USA, 26–28 March 2007; pp. 1–6. [CrossRef]
50. Ouaddah, A.; Abou Elkalam, A.; Ait Ouahman, A. FairAccess: A new Blockchain-based access control framework for the Internet of Things. *Secur. Commun. Netw.* **2016**, *9*, 5943–5964. [CrossRef]
51. Xu, R.; Chen, Y.; Blasch, E.; Chen, G. BlendCAC: A Smart Contract Enabled Decentralized Capability-Based Access Control Mechanism for the IoT. *Computers* **2018**, *7*, 39. [CrossRef]
52. Xu, R.; Chen, Y.; Blasch, E.; Chen, G. BlendCAC: A BLockchain-Enabled Decentralized Capability-Based Access Control for IoTs. In Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, 30 July–3 August 2018; pp. 1027–1034. [CrossRef]
53. Le, T.; Mutka, M.W. CapChain: A Privacy Preserving Access Control Framework Based on Blockchain for Pervasive Environments. In Proceedings of the 2018 IEEE International Conference on Smart Computing (SMARTCOMP), Taormina, Italy, 18–20 June 2018; pp. 57–64. [CrossRef]
54. Choksy, P.; Chaurasia, A.; Rao, U.P.; Kumar, S. Attribute based access control (ABAC) scheme with a fully flexible delegation mechanism for IoT healthcare. *Peer-to-Peer Netw. Appl.* **2023**, *16*, 1445–1467. [CrossRef]
55. Yan, L.; Qin, H.; Yang, K.; Xie, H.; Wang, X.A.; Liu, S. Pairing-Free Certificate-Based Proxy Re-Encryption Plus Scheme for Secure Cloud Data Sharing. *Electronics* **2024**, *13*, 534. [CrossRef]

56. Zemmoudj, S.; Bermad, N.; Omar, M. CA-ADP: Context-Aware Authorization and Delegation Protocol for IoT-based healthcare smart systems. In Proceedings of the 2019 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC), Tunis, Tunisia, 20–22 December 2019; pp. 203–208. [CrossRef]
57. Gusmeroli, S.; Piccione, S.; Rotondi, D. A capability-based security approach to manage access control in the Internet of Things. *Math. Comput. Model.* **2013**, *58*, 1189–1205. [CrossRef]
58. Rabehaja, T.; Pal, S.; Hitchens, M. Design and implementation of a secure and flexible access-right delegation for resource constrained environments. *Future Gener. Comput. Syst.* **2019**, *99*, 593–608. [CrossRef]
59. Hu, Q.; Zheng, G.; Jiang, T. Joint Content and Radio Access for the Internet of Things: A Smart-Contract-Based Trusted Framework. *IEEE Internet Things J.* **2022**, *9*, 18142–18152. [CrossRef]
60. Agyekum, K.O.B.O.; Xia, Q.; Sifah, E.B.; Cobblah, C.N.A.; Xia, H.; Gao, J. A Proxy Re-Encryption Approach to Secure Data Sharing in the Internet of Things Based on Blockchain. *IEEE Syst. J.* **2022**, *16*, 1685–1696. [CrossRef]
61. Andersen, M.P.; Kumar, S.; AbdelBaky, M.; Fierro, G.; Kolb, J.; Kim, H.S.; Culler, D.E.; Popa, R.A. WAVE: A Decentralized Authorization Framework with Transitive Delegation. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1375–1392.
62. Chung, S.H.; Kim, J.H.; Kim, Y. Pragmatic approach using OAuth mechanism for IoT device authorization in cloud. In Proceedings of the 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 12–13 October 2018; pp. 1–4. [CrossRef]
63. Mendy, G.; Ouya, S.; Dioum, I.; Thiaré, O. (Eds.) *e-Infrastructure and e-Services for Developing Countries: 10th EAI International Conference, AFRICOMM 2018, Dakar, Senegal, 29–30 November 2019, Proceedings*; Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering; Springer International Publishing: Cham, Switzerland, 2019; Volume 275. [CrossRef]
64. Chen, X.; Liu, Y.; Chao, H.C.; Li, Y. Ciphertext-Policy Hierarchical Attribute-Based Encryption Against Key-Delegation Abuse for IoT-Connected Healthcare System. *IEEE Access* **2020**, *8*, 86630–86650. [CrossRef]
65. Alphand, O.; Amoretti, M.; Claeys, T.; Dall'Asta, S.; Duda, A.; Ferrari, G.; Rousseau, F.; Tourancheau, B.; Veltri, L.; Zanichelli, F. IoTChain: A blockchain security architecture for the Internet of Things. In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018; pp. 1–6. [CrossRef]
66. Wei, L.; Wu, J.; Long, C.; Li, B. On Designing Context-Aware Trust Model and Service Delegation for Social Internet of Things. *IEEE Internet Things J.* **2021**, *8*, 4775–4787. [CrossRef]
67. Zhang, J.; Yang, Y.; Liu, X.; Ma, J. An Efficient Blockchain-Based Hierarchical Data Sharing for Healthcare Internet of Things. *IEEE Trans. Ind. Inform.* **2022**, *18*, 7139–7150. [CrossRef]
68. Hernández-Ramos, J.L.; Jara, A.J.; Marín, L.; Skarmeta Gómez, A.F. DCapBAC: Embedding authorization logic into smart things through ECC optimizations. *Int. J. Comput. Math.* **2016**, *93*, 345–366. [CrossRef]
69. Li, W.; Jin, C.; Kumari, S.; Xiong, H.; Kumar, S. Proxy re-encryption with equality test for secure data sharing in Internet of Things-based healthcare systems. *Trans. Emerg. Telecommun. Technol.* **2020**, *33*, e3986. [CrossRef]
70. Kumar, D.; Kumar, M. Attribute-Based Data Sharing in Smart Healthcare Environment. In Proceedings of the 2021 4th International Conference on Recent Trends in Computer Science and Technology (ICRTCST), Jamshedpur, India, 11–12 February 2022; pp. 150–157. [CrossRef]
71. Touati, L.; Challal, Y.; Bouabdallah, A. C-CP-ABE: Cooperative Ciphertext Policy Attribute-Based Encryption for the Internet of Things. In Proceedings of the 2014 International Conference on Advanced Networking Distributed Systems and Applications, Bejaia, Algeria, 17–19 June 2014; pp. 64–69. [CrossRef]
72. Lakkundi, V.; Singh, K. Lightweight DTLS implementation in CoAP-based Internet of Things. In Proceedings of the 20th Annual International Conference on Advanced Computing and Communications (ADCOM), Bangalore, India, 19–22 September 2014; pp. 7–11. [CrossRef]
73. Hsieh, H.C.; Chang, K.D.; Wang, L.F.; Chen, J.L.; Chao, H.C. ScriptIoT: A Script Framework for and Internet-of-Things Applications. *IEEE Internet Things J.* **2016**, *3*, 628–636. [CrossRef]
74. Al-Osta, M.; Bali, A.; Gherbi, A. Event driven and semantic based approach for data processing on IoT gateway devices. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 4663–4678. [CrossRef]
75. Hamadi, R.; Khanfor, A.; Ghazzai, H.; Massoud, Y. A Hybrid Artificial Neural Network for Task Offloading in Mobile Edge Computing. In Proceedings of the 2022 IEEE 65th International Midwest Symposium on Circuits and Systems (MWSCAS), Fukuoka, Japan, 7–10 August 2022; pp. 1–4. [CrossRef]
76. Li, C.; Chen, R.; Wang, Y.; Xing, Q.; Wang, B. REEDS: An Efficient Revocable End-to-End Encrypted Message Distribution System for IoT. *IEEE Trans. Dependable Secur. Comput.* **2024**, *21*, 4526–4542. [CrossRef]
77. Shin, Y.A.; Jeong, I.R.; Byun, J.W. Identity-Based Multiproxy Signature with Proxy Signing Key for Internet of Drones. *IEEE Internet Things J.* **2024**, *11*, 4191–4205. [CrossRef]

78. Hu, Y.; Niu, S.; Shao, H. Attribute-based searchable encryption with delegated equality test in cloud-assisted Internet of Things. In Proceedings of the 2022 3rd International Conference on Electronics, Communications and Information Technology (CECIT), Sanya, China, 23–25 December 2022; pp. 328–333. [CrossRef]
79. Saguil, D.; Azim, A. A Layer-Partitioning Approach for Faster Execution of Neural Network-Based Embedded Applications in Edge Networks. *IEEE Access* **2020**, *8*, 59456–59469. [CrossRef]
80. Xu, K.; Wan, Y.; Xue, G. Powering Smart Homes with Information-Centric Networking. *IEEE Commun. Mag.* **2019**, *57*, 40–46. [CrossRef]
81. Bali, A.; Al-Osta, M.; Abdelouahed, G. An Ontology-Based Approach for IoT Data Processing Using Semantic Rules. In *SDL 2017: Model-Driven Engineering for Future Internet*; Lecture Notes in Computer Science; Csöndes, T., Kovács, G., Réthy, G., Eds.; Springer International Publishing: Cham, Switzerland, 2017; Volume 10567, pp. 61–79. [CrossRef]
82. Zhao, Z.; Guo, F.; Wu, G.; Susilo, W.; Wang, B. Secure Infectious Diseases Detection System with IoT-Based e-Health Platforms. *IEEE Internet Things J.* **2022**, *9*, 22595–22607. [CrossRef]
83. Mahamuni, N.; Nikam, H.; Pattewar, G.; Loka, O.; Patil, R. Management of IoT Devices Data Security Using Blockchain and Proxy Re-encryption Algorithm. In *2nd International Conference on Emerging Technologies and Intelligent Systems*; Al-Sharafi, M.A., Al-Emran, M., Al-Kabi, M.N., Shaalan, K., Eds.; Springer International Publishing: Cham, Switzerland, 2023; pp. 551–558.
84. Schnicke, F.; Haque, A.; Kuhn, T.; Espen, D.; Antonino, P.O. Architecture Blueprints to Enable Scalable Vertical Integration of Assets with Digital Twins. In Proceedings of the 2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA), Stuttgart, Germany, 6–9 September 2022; pp. 1–8. [CrossRef]
85. Raza, S.; Seitz, L.; Sitenkov, D.; Selander, G. S3K: Scalable Security with Symmetric Keys—DTLS Key Establishment for the Internet of Things. *IEEE Trans. Autom. Sci. Eng.* **2016**, *13*, 1270–1280. [CrossRef]
86. Albalawi, U.; Joshi, S. Secure and trusted telemedicine in Internet of Things IoT. In Proceedings of the 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, 5–8 February 2018; pp. 30–34. [CrossRef]
87. Kumar, S.; Hu, Y.; Andersen, M.P.; Popa, R.A.; Culler, D.E. JEDI: Many-to-Many End-to-End Encryption and Key Delegation for IoT. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1519–1536.
88. Li, X.; Li, R.; Bai, B.; Zhao, Y.; Liu, G.; Li, R. Outsourcing the Computation of Plaintext Encryption for Homomorphic Encryption. In Proceedings of the 2023 8th International Conference on Computer and Communication Systems (ICCCS), Guangzhou, China, 21–24 April 2023; pp. 408–413. [CrossRef]
89. Jin, Y.; Lee, H. On-Demand Computation Offloading Architecture in Fog Networks. *Electronics* **2019**, *8*, 1076. [CrossRef]
90. Porambage, P.; Braeken, A.; Kumar, P.; Gurtov, A.; Ylianttila, M. CHIP: Collaborative Host Identity Protocol with Efficient Key Establishment for Constrained Devices in Internet of Things. *Wirel. Pers. Commun.* **2017**, *96*, 421–440. [CrossRef]
91. Ben Saied, Y.; Olivereau, A. HIP Tiny Exchange (TEX): A distributed key exchange scheme for HIP-based Internet of Things. In Proceedings of the Third International Conference on Communications and Networking, Hammamet, Tunisia, 29 March–1 April 2012; pp. 1–8. [CrossRef]
92. Denis, M.; Johansen, C.; Jøsang, A. Offline Trusted Device and Proxy Architecture Based on a new TLS Switching Technique. In Proceedings of the 2017 International Workshop on Secure Internet of Things (SIoT), Oslo, Norway, 15 September 2017; pp. 10–19. [CrossRef]
93. Anggorojati, B.; Mahalle, P.N.; Prasad, N.R.; Prasad, R. Capability-based access control delegation model on the federated IoT network. In Proceedings of the The 15th International Symposium on Wireless Personal Multimedia Communications, Taipei, Taiwan, 24–27 September 2012; pp. 604–608.
94. Venkatesan, S.; Rajeshwari, K.R.; Ramakrishnan, M. A Lightweight Searchable Encryption and Delegation Mechanism with Forward Privacy for Improving the Security of Industrial Internet of Things-Cloud Systems. *Res. Sq.* 2022, *preprint*. [CrossRef]
95. Li, H.; Kong, F.; Yu, J.; Zhang, H.; Diao, L.; Tao, Y. Privacy-Preserving and Verifiable Outsourcing Message Transmission and Authentication Protocol in IoT. In Proceedings of the 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Wuhan, China, 9–11 December 2022; pp. 556–564. [CrossRef]
96. Julku, J.; Suomalainen, J.; Kylanpaa, M. Delegated Device Attestation for IoT. In Proceedings of the 2021 8th International Conference on Internet of Things: Systems, Management and Security (IOTSMS), Gandia, Spain, 6–9 December 2021; pp. 1–8. [CrossRef]
97. Sciarretta, G.; Carbone, R.; Ranise, S. A delegated authorization solution for smart-city mobile applications. In Proceedings of the 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), Bologna, Italy, 7–9 September 2016; pp. 1–6. [CrossRef]
98. Kuusijarvi, J. Mitigating loT Security Threats with a Trusted Network Element. In Proceedings of the The 9th International Conference for Internet Technology and Secured Transactions (ICITST-2014), London, UK, 8–10 December 2014.

99. Beltran, V.; Skarmeta, A.F. An overview on delegated authorization for CoAP: Authentication and authorization for Constrained Environments (ACE). In Proceedings of the 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), Reston, VA, USA, 12–14 December 2016; pp. 706–710. [CrossRef]
100. Hummen, R.; Shafagh, H.; Raza, S.; Voig, T.; Wehrle, K. Delegation-based authentication and authorization for the IP-based Internet of Things. In Proceedings of the 2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), Singapore, 30 June–3 July 2014; pp. 284–292. [CrossRef]
101. Kang, N.; Park, J.; Kwon, H.; Jung, S. ESSE: Efficient Secure Session Establishment for Internet-Integrated Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 393754. [CrossRef]
102. Van den Abeele, F.; Moerman, I.; Demeester, P.; Hoebeke, J. Secure Service Proxy: A CoAP(s) Intermediary for a Securer and Smarter Web of Things. *Sensors* **2017**, *17*, 1609. [CrossRef]
103. Atlam, H.F.; Alenezi, A.; Hussein, R.K.; Wills, G.B. Validation of an Adaptive Risk-based Access Control Model for the Internet of Things. *Int. J. Comput. Netw. Inf. Secur. (IJCNIS)* **2018**, *10*, 26–35. [CrossRef]
104. Luo, X.; Yin, L.; Li, C.; Wang, C.; Fang, F.; Zhu, C.; Tian, Z. A Lightweight Privacy-Preserving Communication Protocol for Heterogeneous IoT Environment. *IEEE Access* **2020**, *8*, 67192–67204. [CrossRef]
105. Lee, J.; Oh, J.; Kwon, D.; Kim, M.; Kim, K.; Park, Y. Blockchain-Enabled Key Aggregate Searchable Encryption Scheme for Personal Health Record Sharing with Multi-Delegation. *IEEE Internet Things J.* **2024**, 11, 17482–17494. [CrossRef]
106. Dunnett, K.; Pal, S.; Jadidi, Z.; Jurdak, R. A Blockchain-Based Framework for Scalable and Trustless Delegation of Cyber Threat Intelligence. In Proceedings of the 2023 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Dubai, United Arab Emirates, 1–5 May 2023; pp. 1–9. [CrossRef]
107. Bounaira, S.; Alioua, A.; Souici, I. Blockchain-enabled trust management for secure content caching in mobile edge computing using deep reinforcement learning. *Internet Things* **2024**, *25*, 101081. [CrossRef]
108. Kitchenham, B.; Brereton, O.P.; Budgen, D.; Turner, M.; Bailey, J.; Linkman, S. Systematic literature reviews in software engineering—A systematic literature review. *Inf. Softw. Technol.* **2009**, *51*, 7–15. [CrossRef]
109. Wohlin, C.; Prikladniki, R. Systematic literature reviews in software engineering. *Inf. Softw. Technol.* **2013**, *55*, 919–920. [CrossRef]
110. Seitz, L.; Selander, G.; Gehrmann, C. Authorization framework for the Internet-of-Things. In Proceedings of the 2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), Madrid, Spain, 4–7 June 2013; pp. 1–6. [CrossRef]
111. Bandara, S.; Yashiro, T.; Koshizuka, N.; Sakamura, K. Access control framework for API-enabled devices in smart buildings. In Proceedings of the 2016 22nd Asia-Pacific Conference on Communications (APCC), Yogyakarta, Indonesia, 25–27 August 2016; pp. 210–217. [CrossRef]
112. Kum, S.W.; Kang, M.; Park, J.I. IoT Delegate: Smart Home Framework for Heterogeneous IoT Service Collaboration. *KSII Trans. Internet Inf. Syst.* **2016**, *10*, 3958–3971.
113. Bernal Bernabe, J.; Hernandez Ramos, J.L.; Skarmeta Gomez, A.F. TACIoT: Multidimensional trust-aware access control system for the Internet of Things. *Soft Comput.* **2016**, *20*, 1763–1779. [CrossRef]
114. Hussein, D.; Bertin, E.; Frey, V. A Community-Driven Access Control Approach in Distributed IoT Environments. *IEEE Commun. Mag.* **2017**, *55*, 146–153. [CrossRef]
115. Dao, N.N.; Kim, Y.; Jeong, S.; Park, M.; Cho, S. Achievable Multi-Security Levels for Lightweight IoT-Enabled Devices in Infrastructureless Peer-Aware Communications. *IEEE Access* **2017**, *5*, 26743–26753. [CrossRef]
116. Beltran, V.; Martinez, J.A.; Skarmeta, A.F. User-centric access control for efficient security in smart cities. In Proceedings of the 2017 Global Internet of Things Summit (GIoTS), Geneva, Switzerland, 6–9 June 2017; pp. 1–6. [CrossRef]
117. Chien, H.Y. Group-Oriented Range-Bound Key Agreement for Internet of Things Scenarios. *IEEE Internet Things J.* **2018**, *5*, 1890–1903. [CrossRef]
118. Zhou, Q.; Elbadry, M.; Ye, F.; Yang, Y. Heracles: Scalable, Fine-Grained Access Control for Internet-of-Things in Enterprise Environments. In Proceedings of the IEEE INFOCOM 2018—IEEE Conference on Computer Communications, Honolulu, HI, USA, 15–19 April 2018; pp. 1772–1780. [CrossRef]
119. Behrad, S.; Bertin, E.; Tuffin, S.; Crespi, N. 5G-SSAAC: Slice-specific Authentication and Access Control in 5G. In Proceedings of the 2019 IEEE Conference on Network Softwarization (NetSoft), Paris, France, 24–28 June 2019; pp. 281–285. [CrossRef]
120. Al-Aqrabi, H.; Johnson, A.P.; Hill, R. Dynamic Multiparty Authentication using Cryptographic Hardware for the Internet of Things. In Proceedings of the 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), Leicester, UK, 19–23 August 2019; pp. 21–28. [CrossRef]
121. Lagutin, D.; Kortesniemi, Y.; Fotiou, N.; Siris, V. Enabling Decentralised Identifiers and Verifiable Credentials for Constrained IoT Devices using OAuth-based Delegation. In Proceedings of the Workshop on Decentralized IoT Systems and Security, London, UK, 15 November 2019. [CrossRef]

122. Behrad, S.; Bertin, E.; Tuffin, S.; Crespi, N. A new scalable authentication and access control mechanism for 5G-based IoT. *Future Gener. Comput. Syst.* **2020**, *108*, 46–61. [CrossRef]
123. Xu, H.; He, Q.; Li, X.; Jiang, B.; Qin, K. BDSS-FA: A Blockchain-Based Data Security Sharing Platform with Fine-Grained Access Control. *IEEE Access* **2020**, *8*, 87552–87561. [CrossRef]
124. Hang, L.; Kim, D.H. Reliable Task Management Based on a Smart Contract for Runtime Verification of Sensing and Actuating Tasks in IoT Environments. *Sensors* **2020**, *20*, 1207. [CrossRef]
125. Lin, C.A.; Liao, C.F. User-Managed Access Delegation for Blockchain-driven IoT Services. In Proceedings of the 2020 International Computer Symposium (ICS), Tainan, Taiwan, 17–19 December 2020; pp. 462–467. [CrossRef]
126. Abdel-Malek, M.A.; Akkaya, K.; Bhuyan, A.; Ibrahim, A.S. A Proxy Signature-Based Swarm Drone Authentication with Leader Selection in 5G Networks. *IEEE Access* **2022**, *10*, 57485–57498. [CrossRef]
127. Goyal, G.; Liu, P.; Sural, S. Securing Smart Home IoT Systems with Attribute-Based Access Control. In Proceedings of the 2022 ACM Workshop on Secure and Trustworthy Cyber-Physical Systems, Baltimore, MD, USA, 25–27 April 2022; pp. 37–46. [CrossRef]
128. Jiang, W.; Lin, Z.; Tao, J. An access control scheme for distributed Internet of Things based on adaptive trust evaluation and blockchain. *High-Confid. Comput.* **2023**, *3*, 100104. [CrossRef]
129. Tegane, S.; Semchedine, F.; Boudries, A. An extended Attribute-based access control with controlled delegation in IoT. *J. Inf. Secur. Appl.* **2023**, *76*, 103473. [CrossRef]
130. Alshehri, S.; Bamasaq, O.; Alghazzawi, D.; Jamjoom, A. Dynamic Secure Access Control and Data Sharing Through Trusted Delegation and Revocation in a Blockchain-Enabled Cloud-IoT Environment. *IEEE Internet Things J.* **2023**, *10*, 4239–4256. [CrossRef]
131. Pittaras, I.; Polyzos, G.C. Multi-tenant, Decentralized Access Control for the Internet of Things. In Proceedings of the 2023 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS), Bali, Indonesia, 28–30 November 2023; pp. 28–34. [CrossRef]
132. Vattaparambil Sudarsan, S.; Schelén, O.; Bodin, U. Multilevel Subgranting by Power of Attorney and OAuth Authorization Server in Cyber–Physical Systems. *IEEE Internet Things J.* **2023**, *10*, 15266–15282. [CrossRef]
133. Chen, D.; Zhang, L.; Liao, Z.; Dai, H.N.; Zhang, N.; Shen, X.; Pang, M. Flexible and Fine-Grained Access Control for EHR in Blockchain-Assisted E-Healthcare Systems. *IEEE Internet Things J.* **2024**, *11*, 10992–11007. [CrossRef]
134. Sun, J.; Bao, Y.; Qiu, W.; Lu, R.; Zhang, S.; Guan, Y.; Cheng, X. Privacy-Preserving Fine-Grained Data Sharing with Dynamic Service for the Cloud-Edge IoT. *IEEE Trans. Dependable Secur. Comput.* **2025**, *22*, 1329–1346. [CrossRef]
135. Hernandez-Ramos, J.L.; Pawlowski, M.P.; Jara, A.J.; Skarmeta, A.F.; Ladid, L. Toward a Lightweight Authentication and Authorization Framework for Smart Objects. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 690–702. [CrossRef]
136. Kim, D.Y.; Min, S.D.; Kim, S. A DPN (Delegated Proof of Node) Mechanism for Secure Data Transmission in IoT Services. *Comput. Mater. Contin.* **2019**, *58*, 1–14. [CrossRef]
137. Morales, A.; Robles, T.; Alcarria, R.; Cedeño, E. A Hot-topic based Distribution and Notification of Events in Pub/Sub Mobile Brokers. *Netw. Protoc. Algorithms* **2013**, *5*, 90–110. [CrossRef]
138. Hernández-Ramos, J.; Jara, A.J.; Marin, L.; Skarmeta, A. Distributed Capability-based Access Control for the Internet of Things. *J. Internet Serv. Inf. Secur. (JISIS)* **2013**, *3*, 1–16.
139. Outchakoucht, A.; Es-Samaali, H.; Philippe, J. Dynamic Access Control Policy based on Blockchain and Machine Learning for the Internet of Things. *Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 417–424. [CrossRef]
140. Yang, L.; Zheng, Q.; Fan, X. RSPP: A reliable, searchable and privacy-preserving e-healthcare system for cloud-assisted body area networks. In Proceedings of the IEEE INFOCOM 2017—IEEE Conference on Computer Communications, Atlanta, GA, USA, 1–4 May 2017; pp. 1–9. [CrossRef]
141. Ko, H.; Jin, J.; Keoh, S.L. ViotSOC: Controlling Access to Dynamically Virtualized IoT Services using Service Object Capability. In Proceedings of the 3rd ACM Workshop on Cyber-Physical System Security, Abu Dhabi, United Arab Emirates, 2 April 2017; pp. 69–80. [CrossRef]
142. Tapas, N.; Merlino, G.; Longo, F. Blockchain-Based IoT-Cloud Authorization and Delegation. In Proceedings of the 2018 IEEE International Conference on Smart Computing (SMARTCOMP), Taormina, Sicily, Italy, 18–20 June 2018; pp. 411–416. [CrossRef]
143. Sabrina, F. A Novel Entitlement-based Blockchain-enabled Security Architecture for IoT. In Proceedings of the 2019 29th International Telecommunication Networks and Applications Conference (ITNAC), Auckland, New Zealand, 27–29 November 2019; pp. 1–7. [CrossRef]
144. Porwal, S.; Mittal, S. A Secure Key Delegation Mechanism for Fog Networking. In Proceedings of the 2019 Twelfth International Conference on Contemporary Computing (IC3), Noida, India, 8–10 August 2019; pp. 1–7. [CrossRef]
145. Sun, S.; Chen, S.; Du, R.; Li, W.; Qi, D. Blockchain Based Fine-Grained and Scalable Access Control for IoT Security and Privacy. In Proceedings of the 2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC), Hangzhou, China, 23–25 June 2019; pp. 598–603. [CrossRef]

146. Ali, G.; Ahmad, N.; Cao, Y.; Asif, M.; Cruickshank, H.; Ali, Q.E. Blockchain based permission delegation and access control in Internet of Things (BACI). *Comput. Secur.* **2019**, *86*, 318–334. [CrossRef]
147. Zemmoudj, S.; Bermad, N.; Omar, M. Context-aware pseudonymization and authorization model for IoT-based smart hospitals. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 4473–4490. [CrossRef]
148. Pinjala, S.K.; Sivalingam, K.M. DCACI: A Decentralized Lightweight Capability Based Access Control Framework using IOTA for Internet of Things. In Proceedings of the 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 15–18 April 2019; pp. 13–18. [CrossRef]
149. Xu, L.; Li, J.; Chen, X.; Li, W.; Tang, S.; Wu, H.T. Tc-PEDCKS: Towards time controlled public key encryption with delegatable conjunctive keyword search for Internet of Things. *J. Netw. Comput. Appl.* **2019**, *128*, 11–20. [CrossRef]
150. Deng, H.; Qin, Z.; Sha, L.; Yin, H. A Flexible Privacy-Preserving Data Sharing Scheme in Cloud-Assisted IoT. *IEEE Internet Things J.* **2020**, *7*, 11601–11611. [CrossRef]
151. Nakamura, Y.; Zhang, Y.; Sasabe, M.; Kasahara, S. Exploiting Smart Contracts for Capability-Based Access Control in the Internet of Things. *Sensors* **2020**, *20*, 1793. [CrossRef]
152. Pal, S.; Rabehaja, T.; Hill, A.; Hitchens, M.; Varadharajan, V. On the Integration of Blockchain to the Internet of Things for Enabling Access Right Delegation. *IEEE Internet Things J.* **2020**, *7*, 2630–2639. [CrossRef]
153. Puggioni, E.; Shaghaghi, A.; Doss, R.; Kanhere, S.S. Towards Decentralized IoT Updates Delivery Leveraging Blockchain and Zero-Knowledge Proofs. In Proceedings of the 2020 IEEE 19th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 24–27 November 2020; pp. 1–10. [CrossRef]
154. Ali, G.; Ahmad, N.; Cao, Y.; Khan, S.; Cruickshank, H.; Qazi, E.A.; Ali, A. xDBAuth: Blockchain Based Cross Domain Authentication and Authorization Framework for Internet of Things. *IEEE Access* **2020**, *8*, 58800–58816. [CrossRef]
155. Zhou, X.; Xu, K.; Wang, N.; Jiao, J.; Dong, N.; Han, M.; Xu, H. A Secure and Privacy-Preserving Machine Learning Model Sharing Scheme for Edge-Enabled IoT. *IEEE Access* **2021**, *9*, 17256–17265. [CrossRef]
156. Zhao, J.; Zeng, P.; Choo, K.K.R. An Efficient Access Control Scheme with Outsourcing and Attribute Revocation for Fog-Enabled E-Health. *IEEE Access* **2021**, *9*, 13789–13799. [CrossRef]
157. Zhang, Y.; Liu, W.; Xia, Z.; Wang, Z.; Liu, L.; Zhang, W.; Zhang, H.; Fang, B. Blockchain-Based DNS Root Zone Management Decentralization for Internet of Things. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 6620236. [CrossRef]
158. Ullah, I.; Alkhalifah, A.; Khan, M.A.; Mostafa, S.M. Cost-Effective Proxy Signcryption Scheme for Internet of Things. *Mob. Inf. Syst.* **2021**, *2021*, 2427434. [CrossRef]
159. Sabrina, F.; Jang-Jaccard, J. Entitlement-Based Access Control for Smart Cities Using Blockchain. *Sensors* **2021**, *21*, 5264. [CrossRef]
160. Skarlat, O.; Schulte, S. FogFrame: A framework for IoT application execution in the fog. *PeerJ Comput. Sci.* **2021**, *7*, e588. [CrossRef]
161. Lastname, F.; Lastname, F.; Lastname, F. A Bidirectional Trust Model for Service Delegation in Social Internet of Things. *Future Internet* **2022**, *14*, 135. [CrossRef]
162. Chien, H.Y.; Wang, N.Z. A Novel MQTT 5.0-Based Over-the-Air Updating Architecture Facilitating Stronger Security. *Electronics* **2022**, *11*, 3899. [CrossRef]
163. Cha, H.J.; Yang, H.K.; Song, Y.J. A Study on Vehicle Monitoring Service Using Attribute-Based Security Scheme in Cyber–Physical Systems. *Appl. Sci.* **2022**, *12*, 4300. [CrossRef]
164. Li, C.; Li, F.; Huang, C.; Yin, L.; Luo, T.; Wang, B. A Traceable Capability-based Access Control for IoT. *Comput. Mater. Contin.* **2022**, *72*, 4967–4982. [CrossRef]
165. Ali, A.; Almaiah, M.A.; Hajjej, F.; Pasha, M.F.; Fang, O.H.; Khan, R.; Teo, J.; Zakarya, M. An Industrial IoT-Based Blockchain-Enabled Secure Searchable Encryption Approach for Healthcare Systems Using Neural Network. *Sensors* **2022**, *22*, 572. [CrossRef]
166. Bayreuther, S.; Jacob, F.; Grotz, M.; Kartmann, R.; Peller-Konrad, F.; Paus, F.; Hartenstein, H.; Asfour, T. BlueSky: Combining Task Planning and Activity-Centric Access Control for Assistive Humanoid Robots. In Proceedings of the 27th ACM on Symposium on Access Control Models and Technologies, New York, NY, USA, 8–10 June 2022; pp. 185–194. [CrossRef]
167. Xiong, H.; Wang, L.; Zhou, Z.; Zhao, Z.; Huang, X.; Kumari, S. Burn After Reading: Adaptively Secure Puncturable Identity-Based Proxy Re-Encryption Scheme for Securing Group Message. *IEEE Internet Things J.* **2022**, *9*, 11248–11260. [CrossRef]
168. Latif, R. ConTrust: A Novel Context-Dependent Trust Management Model in Social Internet of Things. *IEEE Access* **2022**, *10*, 46526–46537. [CrossRef]
169. Xue, L. DSAS: A Secure Data Sharing and Authorized Searchable Framework for e-Healthcare System. *IEEE Access* **2022**, *10*, 30779–30791. [CrossRef]
170. Pinjala, S.K.; Vivek, S.S.; Sivalingam, K.M. Delegated Anonymous Credentials with Revocation Capability for IoT Service Chains (*DANCIS*). *IEEE Internet Things J.* **2022**, *9*, 3729–3742. [CrossRef]
171. Silva, C.; Barraca, J.P. Dynamic Delegation-based Privacy Preserving in IoT Architectures. In Proceedings of the 2022 9th International Conference on Future Internet of Things and Cloud (FiCloud), Rome, Italy, 22–24 August 2022; pp. 46–54. [CrossRef]

172. Banerjee, A.; Sufian, A.; Paul, K.K.; Gupta, S.K. EDTP: Energy and Delay Optimized Trajectory Planning for UAV-IoT Environment. *Comput. Netw.* **2022**, *202*, 108623. [CrossRef]
173. Qushtom, H.; Misic, J.; Misic, V.B. Efficient multi-tier, multiple entry PBFT consensus algorithm for IoT. In Proceedings of the ICC 2022—IEEE International Conference on Communications, Seoul, Republic of Korea, 16–20 May 2022; pp. 44–49. [CrossRef]
174. Heo, J.; Jang, H.; Lee, H. How to divide a permission token in the delegation process of blockchain-based access control for IoT. In Proceedings of the 2022 IEEE International Systems Conference (SysCon), Montreal, QC, Canada, 25–28 April 2022; pp. 1–8. [CrossRef]
175. Ju, C.; Tang, W.; Chenli, C.; Lee, G.; Seo, J.H.; Jung, T. Monitoring Provenance of Delegated Personal Data with Blockchain. In Proceedings of the 2022 IEEE International Conference on Blockchain (Blockchain), Espoo, Finland, 22–25 August 2022; pp. 11–20. [CrossRef]
176. Panneerselvam, N.; Krithiga, S. Mutual-contained access delegation scheme for the Internet of Things user services. *Distrib. Parallel Databases* **2022**, *40*, 835–860. [CrossRef]
177. Wang, J.; Chow, S.S.M. Omnes pro uno: Practical Multi-Writer Encrypted Database. In Proceedings of the 31st USENIX Security Symposium (USENIX Security 22), Boston, MA, USA, 10–12 August 2022; pp. 2371–2388.
178. Sama, M.R.; Kiess, W.; Guerzoni, R.; Thakolsri, S.; Jurjens, J. Redefining the Trust Model for the Internet of Everything in the 6G era. In Proceedings of the 2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Kyoto, Japan, 12–15 September 2022; pp. 1400–1406. [CrossRef]
179. Shur, D.; Di Crescenzo, G.; Zhang, Q.; Chen, T.; Krishnan, R.; Lin, Y.J.; Patni, Z.; Alexander, S.; Tsudik, G. SEDIMENT: An IoT-device-centric Methodology for Scalable 5G Network Security. In Proceedings of the 2022 IEEE Wireless Communications and Networking Conference (WCNC), Austin, TX, USA, 10–13 April 2022; pp. 49–54. [CrossRef]
180. Tanyingyong, V.; Olsson, R.; Hidell, M.; Sjodin, P. Scalable IoT Sensing Systems with Dynamic Sinks. *IEEE Internet Things J.* **2022**, *9*, 7211–7227. [CrossRef]
181. Goh, Y.; Yun, J.; Jung, D.; Chung, J.M. Secure Trust-Based Delegated Consensus for Blockchain Frameworks Using Deep Reinforcement Learning. *IEEE Access* **2022**, *10*, 118498–118511. [CrossRef]
182. Qiao, Z.; Zhou, Y.; Yang, B.; Zhang, M.; Wang, T.; Xia, Z. Secure and Efficient Certificate-Based Proxy Signature Schemes for Industrial Internet of Things. *IEEE Syst. J.* **2022**, *16*, 4719–4730. [CrossRef]
183. Rios, E.; Higuero, M.; Larrucea, X.; Rak, M.; Casola, V.; Iturbe, E. Security and Privacy Service Level Agreement composition for Internet of Things systems on top of standard controls. *Comput. Electr. Eng.* **2022**, *98*, 107690. [CrossRef]
184. Wanniarachchi, S.T.; Turau, V. A Fault-Tolerant Distributed Air-to-Ground Communication Architecture for Urban Air Mobility. In Proceedings of the 2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT), Pafos, Cyprus, 19–21 June 2023; pp. 639–646. [CrossRef]
185. Wang, C.; Wu, B. A Linear Homomorphic Proxy Signature Scheme Based on Blockchain for Internet of Things. *Comput. Model. Eng. Sci.* **2023**, *136*, 1857–1878. [CrossRef]
186. Alharbi, A. Applying Access Control Enabled Blockchain (ACE-BC) Framework to Manage Data Security in the CIS System. *Sensors* **2023**, *23*, 3020. [CrossRef]
187. Malhotra, A. Blend CAC: Integration for the Blockchain for Distributed Potential Network Access for the Internet of Things. In Proceedings of the 2023 International Conference on Artificial Intelligence and Smart Communication (AISC), Greater Noida, India, 27–29 January 2023; pp. 1145–1149. [CrossRef]
188. Lu, Y.; Wang, D.; Obaidat, M.S.; Vijayakumar, P. Edge-Assisted Intelligent Device Authentication in Cyber–Physical Systems. *IEEE Internet Things J.* **2023**, *10*, 3057–3070. [CrossRef]
189. Trivedi, H.S.; Patel, S.J. Key-aggregate searchable encryption with multi-user authorization and keyword untraceability for distributed IoT healthcare systems. *Trans. Emerg. Telecommun. Technol.* **2023**, *34*, e4734. [CrossRef]
190. Sahraoui, S.; Henni, N. SAMP-RPL: Secure and adaptive multipath RPL for enhanced security and reliability in heterogeneous IoT-connected low power and lossy networks. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 409–429. [CrossRef]
191. Hamad, M.; Finkenzeller, A.; Liu, H.; Lauinger, J.; Prevelakis, V.; Steinhorst, S. SEEMQTT: Secure End-to-End MQTT-Based Communication for Mobile IoT Systems Using Secret Sharing and Trust Delegation. *IEEE Internet Things J.* **2023**, *10*, 3384–3406. [CrossRef]
192. Seifhosseini, S.; Hosseini Shirvani, M.; Ramzanpoor, Y. Multi-objective cost-aware bag-of-tasks scheduling optimization model for IoT applications running on heterogeneous fog environment. *Comput. Netw.* **2024**, *240*, 110161. [CrossRef]
193. Fan, X.; Chai, Q. Roll-DPoS: A Randomized Delegated Proof of Stake Scheme for Scalable Blockchain-Based Internet of Things Systems. In Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, New York, NY, USA, 5–7 November 2018; pp. 482–484. [CrossRef]
194. Maselli, G.; Piva, M.; Restuccia, F. HyBloSE: Hybrid blockchain for secure-by-design smart environments. In Proceedings of the 3rd Workshop on Cryptocurrencies and Blockchains for Distributed Systems, London UK, 25 September 2020; pp. 23–28. [CrossRef]

195. Vairagade, R.S.; Brahmananda, S.H. Enabling machine learning-based side-chaining for improving QoS in blockchain-powered IoT networks. *Trans. Emerg. Telecommun. Technol.* **2021**, *33*, e4433. [CrossRef]
196. Wang, Y.; Che, T.; Zhao, X.; Zhou, T.; Zhang, K.; Hu, X. A Blockchain-Based Privacy Information Security Sharing Scheme in Industrial Internet of Things. *Sensors* **2022**, *22*, 3426. [CrossRef]
197. Huang, X.; Zhang, Y.; Li, D.; Han, L. A Solution for Bilayer Energy-Trading Management in Microgrids Using Multiblockchain. *IEEE Internet Things J.* **2022**, *9*, 13886–13900. [CrossRef]
198. Ahmed, I.; Zhang, Y.; Jeon, G.; Lin, W.; Khosravi, M.R.; Qi, L. A blockchain- and artificial intelligence-enabled smart IoT framework for sustainable city. *Int. J. Intell. Syst.* **2022**, *37*, 6493–6507. [CrossRef]
199. Wang, P.; Zhao, Y.; Obaidat, M.S.; Wei, Z.; Qi, H.; Lin, C.; Xiao, Y.; Zhang, Q. Blockchain-Enhanced Federated Learning Market with Social Internet of Things. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 3405–3421. [CrossRef]
200. Wang, H.; Jiang, H.; Wu, J.; Zhou, P.; Huang, H. Blockchain-Governed Federated Transfer Learning for Secure Internet of Drones Networks. *IEEE Internet Things Mag.* **2022**, *5*, 134–139. [CrossRef]
201. Zhi, H.; Ge, H.; Wang, Y. Cooperative Communication Method Based on Block Chain for a Large Number of Distributed Terminals. *IEEE Access* **2022**, *10*, 11679–11695. [CrossRef]
202. Kaur, M.; Gupta, S.; Kumar, D.; Verma, C.; Neagu, B.C.; Raboaca, M.S. Delegated Proof of Accessibility (DPoAC): A Novel Consensus Protocol for Blockchain Systems. *Mathematics* **2022**, *10*, 2336. [CrossRef]
203. Geng, T.; Njilla, L.; Huang, C.T. Delegated Proof of Secret Sharing: A Privacy-Preserving Consensus Protocol Based on Secure Multiparty Computation for IoT Environment. *Network* **2022**, *2*, 66–80. [CrossRef]
204. Misic, J.; Misic, V.B.; Chang, X. Delegated Proof of Stake Consensus with Mobile Voters and Multiple Entry PBFT Voting. In Proceedings of the GLOBECOM 2022—2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 4–8 December 2022; pp. 6253–6258. [CrossRef]
205. Yang, L.; Zou, W.; Wang, J.; Tang, Z. EdgeShare: A blockchain-based edge data-sharing framework for Industrial Internet of Things. *Neurocomputing* **2022**, *485*, 219–232. [CrossRef]
206. Ledwaba, L.P.; Hancke, G.P.; Isaac, S.J. Investigating Distance Bounding for Delegated Proof-of-Proximity Consensus within IIoT. In Proceedings of the 2022 IEEE 31st International Symposium on Industrial Electronics (ISIE), Anchorage, AK, USA, 1–3 June 2022; pp. 627–630. [CrossRef]
207. Hu, D.; Chen, J.; Zhou, H.; Yu, K.; Qian, B.; Xu, W. Leveraging Blockchain for Multi-Operator Access Sharing Management in Internet of Vehicles. *IEEE Trans. Veh. Technol.* **2022**, *71*, 2774–2787. [CrossRef]
208. Bachani, V.; Bhattacharjya, A. Preferential Delegated Proof of Stake (PDPoS)—Modified DPoS with Two Layers towards Scalability and Higher TPS. *Symmetry* **2022**, *15*, 4. [CrossRef]
209. Cui, J.; Ouyang, F.; Ying, Z.; Wei, L.; Zhong, H. Secure and Efficient Data Sharing Among Vehicles Based on Consortium Blockchain. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 8857–8867. [CrossRef]
210. Sasikumar, A.; Ravi, L.; Kotecha, K.; Saini, J.R.; Varadarajan, V.; Subramaniyaswamy, V. Sustainable Smart Industry: A Secure and Energy Efficient Consensus Mechanism for Artificial Intelligence Enabled Industrial Internet of Things. *Comput. Intell. Neurosci.* **2022**, *2022*, 1419360. [CrossRef]
211. Wang, Y.; Su, Z.; Xu, Q.; Li, R.; Luan, T.H.; Wang, P. A Secure and Intelligent Data Sharing Scheme for UAV-Assisted Disaster Rescue. *IEEE/ACM Trans. Netw.* **2023**, *31*, 2422–2438. [CrossRef]
212. Namasudra, S.; Sharma, P. Achieving a Decentralized and Secure Cab Sharing System Using Blockchain Technology. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 15568–15577. [CrossRef]
213. Zhao, J.; Hu, H.; Huang, F.; Guo, Y.; Liao, L. Authentication Technology in Internet of Things and Privacy Security Issues in Typical Application Scenarios. *Electronics* **2023**, *12*, 1812. [CrossRef]
214. Wang, J.; Lin, X.; Wu, Y.; Wu, J. Blockchain-Enabled Lightweight Fine-Grained Searchable Knowledge Sharing for Intelligent IoT. *IEEE Internet Things J.* **2023**, *10*, 21566–21579. [CrossRef]
215. Razaque, A.; Yoo, J.; Bektemyssova, G.; Alshammari, M.; Chinibayeva, T.T.; Amanzholova, S.; Alotaibi, A.; Umutkulov, D. Efficient Internet-of-Things Cyberattack Depletion Using Blockchain-Enabled Software-Defined Networking and 6G Network Technology. *Sensors* **2023**, *23*, 9690. [CrossRef]
216. Alghamdi, S.; Albeshri, A.; Alhusayni, A. Enabling a Secure IoT Environment Using a Blockchain-Based Local-Global Consensus Manager. *Electronics* **2023**, *12*, 3721. [CrossRef]
217. Cullen, A.; Zhao, L.; Vigneri, L.; Shorten, R. Improving Quality of Service for Users of Leaderless DAG-Based Distributed Ledgers. *Distrib. Ledger Technol. Res. Pract.* **2023**, *2*, 1–18. [CrossRef]
218. Qi, L.; Tian, J.; Chai, M.; Cai, H. LightPoW: A trust based time-constrained PoW for blockchain in Internet of Things. *Comput. Netw.* **2023**, *220*, 109480. [CrossRef]
219. Kaur, M.; Gupta, S. Optimization of a Consensus Protocol in Blockchain-IoT Convergence. In Proceedings of the 2023 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 1–3 March 2023; pp. 1–5.

220. Yang, F.; Xu, F.; Feng, T.; Qiu, C.; Zhao, C. pDPoSt+sPBFT: A High Performance Blockchain-Assisted Parallel Reinforcement Learning in Industrial Edge-Cloud Collaborative Network. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 2744–2759. [CrossRef]
221. Mingjie Zhao, M.Z.; Mingjie Zhao, C.D.; Cheng Dai, B.G. Safe and Efficient Delegated Proof of Stake Consensus Mechanism Based on Dynamic Credit in Electronic Transaction. *J. Internet Technol.* **2023**, *24*, 123–133. [CrossRef]
222. Wang, L.; Zheng, Y.; Zhang, Y.; Li, F. Secure Spectrum Sharing for Satellite Internet-of-Things Based on Blockchain. *Wirel. Pers. Commun.* **2023**, *131*, 357–369. [CrossRef]
223. Xiao, N.; Wang, Z.; Sun, X.; Miao, J. A novel blockchain-based digital forensics framework for preserving evidence and enabling investigation in industrial Internet of Things. *Alex. Eng. J.* **2024**, *86*, 631–643. [CrossRef]
224. Lin, L.; Wu, J.; Zhou, Z.; Zhao, J.; Li, P.; Xiong, J. Computing Power Networking Meets Blockchain: A Reputation-Enhanced Trading Framework for Decentralized IoT Cloud Services. *IEEE Internet Things J.* **2024**, *11*, 17082–17096. [CrossRef]
225. Mohammed, M.A.; Wahab, H.B.A. Enhancing IoT Data Security with Lightweight Blockchain and Okamoto Uchiyama Homomorphic Encryption. *Comput. Model. Eng. Sci.* **2024**, *138*, 1731–1748. [CrossRef]
226. Tsang, Y.P.; Lee, C.K.M.; Zhang, K.; Wu, C.H.; Ip, W.H. On-Chain and Off-Chain Data Management for Blockchain-Internet of Things: A Multi-Agent Deep Reinforcement Learning Approach. *J. Grid Comput.* **2024**, *22*, 16. [CrossRef]
227. Al-Marridi, A.Z.; Mohamed, A.; Erbad, A. Optimized blockchain-based healthcare framework empowered by mixed multi-agent reinforcement learning. *J. Netw. Comput. Appl.* **2024**, *224*, 103834. [CrossRef]
228. Jiang, L.; Liu, Y.; Tian, H.; Tang, L.; Xie, S. Resource Efficient Federated Learning and DAG Blockchain with Sharding in Digital Twin Driven Industrial IoT. *IEEE Internet Things J.* **2024**, *11*, 17113–17127. [CrossRef]
229. Wang, Y.; Xing, X.; Li, P.; Wang, G. LT-DBFT: A Hierarchical Blockchain Consensus Using Location and Trust in IoT. *IEEE Internet Things J.* 2025, *early access*. [CrossRef]
230. Leo, M.; Battisti, F.; Carli, M.; Neri, A. A federated architecture approach for Internet of Things security. In Proceedings of the 2014 Euro Med Telco Conference (EMTC), Naples, Italy, 12–15 November 2014; pp. 1–5. [CrossRef]
231. Porambage, P.; Braeken, A.; Kumar, P.; Gurtov, A.; Ylianttila, M. Efficient Key Establishment for Constrained IoT Devices with Collaborative HIP-Based Approach. In Proceedings of the 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, USA, 6–10 December 2015; pp. 1–6. [CrossRef]
232. Xu, P.; Tang, X.; Wang, W.; Jin, H.; Yang, L.T. Fast and Parallel Keyword Search Over Public-Key Ciphertexts for Cloud-Assisted IoT. *IEEE Access* **2017**, *5*, 24775–24784. [CrossRef]
233. Huang, Q.; Wang, L.; Yang, Y. DECENT: Secure and fine-grained data access control with policy updating for constrained IoT devices. *World Wide Web* **2018**, *21*, 151–167. [CrossRef]
234. Hao, J.; Huang, C.; Liu, J.; Xian, M.; Shen, X. Efficient Outsourced Data Access Control with User Revocation for Cloud-Based IoT. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 9–13 December 2018; pp. 1–6. [CrossRef]
235. Nguyen, K.T.; Oualha, N.; Laurent, M. Securely outsourcing the ciphertext-policy attribute-based encryption. *World Wide Web* **2018**, *21*, 169–183. [CrossRef]
236. Moor, L.; Bitter, L.; Prado, M.D.; Pazos, N.; Ouerhani, N. IoT meets distributed AI—Deployment scenarios of Bonseyes AI applications on FIWARE. In Proceedings of the 2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC), London, UK, 29–31 October 2019; pp. 1–2. [CrossRef]
237. Elhabob, R.; Zhao, Y.; Eltayieb, N.; Abdelgader, A.M.S.; Xiong, H. Identity-based encryption with authorized equivalence test for cloud-assisted IoT. *Clust. Comput.* **2020**, *23*, 1085–1101. [CrossRef]
238. Wang, W.; Xu, P.; Liu, D.; Yang, L.T.; Yan, Z. Lightweighted Secure Searching Over Public-Key Ciphertexts for Edge-Cloud-Assisted Industrial IoT Devices. *IEEE Trans. Ind. Inform.* **2020**, *16*, 4221–4230. [CrossRef]
239. Xu, S.; Ning, J.; Li, Y.; Zhang, Y.; Xu, G.; Huang, X.; Deng, R. Match in My Way: Fine-Grained Bilateral Access Control for Secure Cloud-Fog Computing. *IEEE Trans. Dependable Secur. Comput.* **2020**, *19*, 1064–1077. [CrossRef]
240. Zheng, Y.; Lu, R.; Mamun, M. Privacy-Preserving Computation Offloading for Time-Series Activities Classification in eHealthcare. In Proceedings of the ICC 2020—2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 1–6. [CrossRef]
241. Gochhayat, S.P.; Lal, C.; Sharma, L.; Sharma, D.P.; Gupta, D.; Saucedo, J.A.M.; Kose, U. Reliable and secure data transfer in IoT networks. *Wirel. Netw.* **2020**, *26*, 5689–5702. [CrossRef]
242. Tao, Y.; Xu, P.; Jin, H. Secure Data Sharing and Search for Cloud-Edge-Collaborative Storage. *IEEE Access* **2020**, *8*, 15963–15972. [CrossRef]
243. Yi, C.; Cai, J.; Zhu, K.; Wang, R. A Queueing Game Based Management Framework for Fog Computing with Strategic Computing Speed Control. *IEEE Trans. Mob. Comput.* **2022**, *21*, 1537–1551. [CrossRef]
244. Wang, L.; Tian, Y.; Xiong, J. Achieving reliable and anti-collusive outsourcing computation and verification based on blockchain in 5G-enabled IoT. *Digit. Commun. Netw.* **2022**, *8*, 644–653. [CrossRef]

245. De Hoz Diego, J.D.; Temperekidis, A.; Katsaros, P.; Konstantinou, C. An IoT Digital Twin for Cyber-Security Defence Based on Runtime Verification. In *Leveraging Applications of Formal Methods, Verification and Validation. Verification Principles*; Lecture Notes in Computer Science; Margaria, T., Steffen, B., Eds.; Springer International Publishing: Cham, Switzerland, 2022; Volume 13701, pp. 556–574. [CrossRef]
246. Kim, W.B.; Kim, S.H.; Seo, D.; Lee, I.Y. Certificateless Group to Many Broadcast Proxy Reencryptions for Data Sharing towards Multiple Parties in IoTs. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 1903197. [CrossRef]
247. Jiang, L.; Qin, Z. Edge-assisted Puncturable Fine-grained Task Distribution for the IoT-oriented Crowdsensing. In Proceedings of the 2022 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing & Communications (GreenCom) and IEEE Cyber, Physical & Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), Espoo, Finland, 22–25 August 2022; pp. 131–138. [CrossRef]
248. Zhou, J.; Cao, Z.; Dong, X.; Choo, K.K.R. Efficient Privacy-Preserving Outsourced Discrete Wavelet Transform in the Encrypted Domain. *IEEE Trans. Cloud Comput.* **2022**, *10*, 366–382. [CrossRef]
249. Liu, D.; Li, Z.; Wang, C.; Ren, Y. Enabling secure mutual authentication and storage checking in cloud-assisted IoT. *Math. Biosci. Eng.* **2022**, *19*, 11034–11046. [CrossRef]
250. Cai, G.; Wei, X.; Yao, L. Privacy-preserving CNN feature extraction and retrieval over medical images. *Int. J. Intell. Syst.* **2022**, *37*, 9267–9289. [CrossRef]
251. Chen, Y.C.; Chang, C.C.; Hung, C.C.; Lin, J.F.; Hsu, S.Y. SecDT: Privacy-Preserving Outsourced Decision Tree Classification without Polynomial Forms in Edge-Cloud Computing. *IEEE Trans. Signal Inf. Process. Netw.* **2022**, *8*, 1037–1048. [CrossRef]
252. Li, W.; Xia, C.; Wang, C.; Wang, T. Secure and Temporary Access Delegation with Equality Test for Cloud-Assisted IoV. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 20187–20201. [CrossRef]
253. Benomar, Z.; Campobello, G.; Segreto, A.; Battaglia, F.; Longo, F.; Merlino, G.; Puliafito, A. A Fog-Based Architecture for Latency-Sensitive Monitoring Applications in Industrial Internet of Things. *IEEE Internet Things J.* **2023**, *10*, 1908–1918. [CrossRef]
254. Meng, X.; Yang, C.; Qi, Y.; Liang, W.; Xu, Z.; Li, K.; Deng, H. A Novel Multi-Party Authentication Scheme for FCN-based MIoT Systems in Natural Language Processing Environment. In *ACM Transactions on Asian and Low-Resource Language Information Processing*; Association for Computing Machinery: New York, NY, USA, 2023; p. 3590149. [CrossRef]
255. Yao, M.; Gan, Q.; Wang, X.; Yang, Y. A key-insulated secure multi-server authenticated key agreement protocol for edge computing-based VANETs. *Internet Things* **2023**, *21*, 100679. [CrossRef]
256. Chouikhi, S.; Esseghir, M.; Merghem-Boulahia, L. Computation Offloading for Industrial Internet of Things: A Cooperative Approach. In Proceedings of the 2023 International Wireless Communications and Mobile Computing (IWCMC), Marrakesh, Morocco, 19–23 June 2023; pp. 626–631. [CrossRef]
257. Song, Z.; Ma, H.; Zhang, R.; Xu, W.; Li, J. Everything Under Control: Secure Data Sharing Mechanism for Cloud-Edge Computing. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 2234–2249. [CrossRef]
258. Atutxa, A.; Astorga, J.; Barcelo, M.; Urbieta, A.; Jacob, E. Improving efficiency and security of IIoT communications using in-network validation of server certificate. *Comput. Ind.* **2023**, *144*, 103802. [CrossRef]
259. Kurt, A.; Akkaya, K.; Yilmaz, S.; Mercan, S. LNGate^2: Secure Bidirectional IoT Micro-Payments Using Bitcoin's Lightning Network and Threshold Cryptography. *IEEE Trans. Mob. Comput.* **2023**, *23*, 6027–6044. [CrossRef]
260. Wang, L.; Huang, C.; Cheng, H. Novel proxy signature from lattice for the post-quantum internet of things. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 9939–9946. [CrossRef]
261. Srikanth, K.; Rajesh, P.G.; Prasad, N.D.; Asmathulla, M.; Reddy, T.P.K. Proxy-Based Re-Encryption Design for the IoT Ecosystem. In Proceedings of the 2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India, 25–26 May 2023; pp. 1–6. [CrossRef]
262. Wu, T.; Ma, X.; Zhang, C.; Liu, X.; Yang, G.; Zhu, L. Towards Fine-Grained Task Allocation with Bilateral Access Control for Intelligent Transportation Systems. *IEEE Internet Things J.* **2023**, *11*, 14814–14828. [CrossRef]
263. Sun, Y.; Du, X.; Niu, S.; Zhou, S. A lightweight attribute-based signcryption scheme based on cloud-fog assisted in smart healthcare. *PLoS ONE* **2024**, *19*, e0297002. [CrossRef] [PubMed]
264. Routray, K.; Bera, P. Fog-Assisted Dynamic IoT Device Access Management Using Attribute-Based Encryption. In Proceedings of the 25th International Conference on Distributed Computing and Networking, Chennai, India, 4–7 January 2024; pp. 346–352. [CrossRef]
265. Qu, Z.; Meng, Y.; Liu, B.; Muhammad, G.; Tiwari, P. QB-IMD: A Secure Medical Data Processing System with Privacy Protection Based on Quantum Blockchain for IoMT. *IEEE Internet Things J.* **2024**, *11*, 40–49. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Review

Health IoT Threats: Survey of Risks and Vulnerabilities

Samaneh Madanian [1,*], Tserendorj Chinbat [1], Maduka Subasinghage [2], David Airehrour [3], Farkhondeh Hassandoust [4] and Sira Yongchareon [1]

[1] Department of Computer Science and Software Engineering, Auckland University of Technology (AUT), 6 St. Paul Street, Auckland 1010, New Zealand; tserendorjc9@gmail.com (T.C.); sira.yongchareon@aut.ac.nz (S.Y.)
[2] Business School, University of Western Australia, Perth 6009, Australia; maduka.subasinghage@uwa.edu.au
[3] Together Communications, 77 Cook Street, Auckland 1010, New Zealand; david@wearetogether.co.nz
[4] Department of Information Systems and Operation Management, University of Auckland, 12 Grafton Road, Auckland CBD, Auckland 1010, New Zealand; farkhondeh.hassandoust@auckland.ac.nz
* Correspondence: sam.madanian@aut.ac.nz

Abstract: The secure and efficient collection of patients' vital information is a challenge faced by the healthcare industry. Through the adoption and application of Internet of Things (IoT), the healthcare industry has seen an improvement in the quality of delivered services and patient safety. However, IoT utilization in healthcare is challenging due to the sensitive nature of patients' clinical information and communicating this across heterogeneous networks and among IoT devices. We conducted a semi-systematic literature review to provide an overview of IoT security and privacy challenges in the healthcare sector over time. We collected 279 studies from 5 scientific databases, of which 69 articles met the requirements for inclusion. We performed thematic and qualitative content analysis to extract trends and information. According to our analysis, the vulnerabilities in IoT in healthcare are classified into three main layers: perception, network, and application. We comprehensively reviewed IoT privacy and security threats on each layer. Different technological advancements were suggested to address the identified vulnerabilities in healthcare. This review has practical implications, emphasizing that healthcare organizations, software developers, and device manufacturers must prioritize healthcare IoT security and privacy. A comprehensive, multilayered security approach, security-by-design principles, and training for staff and end-users must be adopted. Regulators and policy makers must also establish and enforce standards and regulations that promote the security and privacy of healthcare IoT. Overall, this study underscores the importance of ensuring the security and privacy of healthcare IoT, with stakeholders' coordinated efforts to address the complex and evolving security and privacy threats in this field. This can enhance healthcare IoT trust and reliability, reduce the risks of security and privacy issues and attacks, and ultimately improve healthcare delivery quality and safety.

Keywords: internet of things; digital health; IoT; security; privacy; healthcare

1. Introduction

Collection of patient's vital information is a major challenge faced by the healthcare industry [1], particularly in real-time settings. To overcome this challenge, the adoption of Internet of Things (IoT) technologies has been substantially increased in mainstream healthcare [2] and disaster situations [3]. IoT refers to a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies [4]. IoT embraces sensing devices and technologies such as sensors, RFID, GPS, and infrared sensors [5] to automatically gather various formats of data such as movements, sound, heat, moisture, light, and location. IoT has been applied in healthcare for various purposes, including remote patient monitoring [6], telemedicine [7], Electronic Healthcare

Record (EHR) management [8], and smart ambulance services [9]. The use of IoT devices has enabled medical practitioners to receive real-time information regarding patients' conditions, which facilitates health monitoring and treatment [10]. The integration of IoT in healthcare has also led to increased efficiency in the delivery of healthcare services [11] and improved patient outcomes [12].

IoT architecture is comprised of three main layers [5] (as shown in Figure 1):

The perception layer: Also referred to as the sensing layer [13], this layer integrates hardware for perception and data collection. Objects/human identification and environmental condition sensing are performed through various IoT devices in this layer [13,14]. With advancements in IoT technologies, this layer has become more sophisticated, including the use of sensors, cameras, and wearable devices to capture and transmit data [15,16].

The network layer: This layer is responsible for providing secure and dependable data transmission support and aggregating data from multiple sources [17,18]. Data transmission in this layer occurs through different networks, such as mobile communication [19], satellite [20], wireless [21], and the internet [21].

The application layer: This layer enables user interaction with IoT devices based on their needs and requirements [22]. This is crucial in the IoT architecture as it provides the platform for delivering IoT services and applications [23,24]. With the integration of artificial intelligence (AI) and machine learning algorithms, the application layer has become more intelligent, capable of providing personalized experiences to users and making real-time decisions [25].

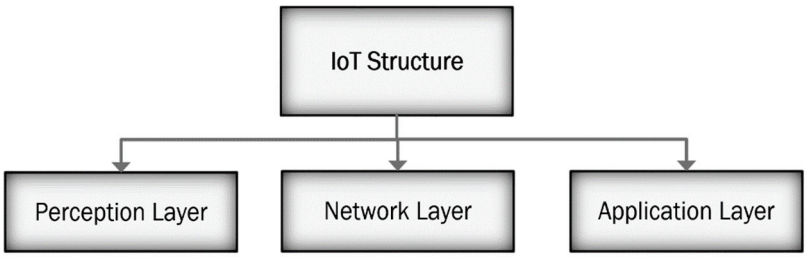


Figure 1. IoT structure.

The rise of IoT in healthcare has been a game changer for medical services, revolutionizing patient treatment with the deployment of online monitoring, usage of wearable devices, and remote access to medical consultations. Nevertheless, with the advancement of IoT technologies, safeguarding sensitive health information has become a major challenge and a concern [2,3,26,27]. As clinical data are being captured, transferred, and archived via these IoT medical devices, the volume of cyber incidents, data breaches, and infringements of privacy regarding such devices is also rising. This research is motivated by the lack of a grounded understanding of the specific risks, threats, and vulnerabilities posed by using IoT devices in the healthcare sector [16,28,29]. The use of IoT devices has made it easier for hackers to obtain access to sensitive patient information such as patient's email accounts, passwords, and medical records [29] and manipulate those records, which can lead to serious implications for patient's health [30]. Moreover, the leakage of patient's sensitive information can cause financial threats, mental anguish, medical identity theft, and possible social stigma or discrimination against people with a particular disease [31,32].

This study is further justified by the increase in the frequency and severity of hacking activities that have adversely impacted the healthcare industry [32–35]. Over the last decade, healthcare institutions have seen a dramatic rise in data breaches, which are underpinned by unauthorized access to millions of confidential patient information. Health-related data are probably the most sensitive and vulnerable to threats, as they not only include medical records and treatment protocols but personal information, including Social Security numbers, genetic data, and health insurance information as well. By accentuating the weakness of each layer, this research study lays the groundwork for further studies and

practical measures directed towards the improvement of the security and privacy of IoT healthcare devices for safe use and efficiency in enhancing patient care.

The remaining sections are structured as follows: A comprehensive introduction to the main concepts behind IoT in healthcare and the security and privacy concerns are provided in Section 2. This study's methodology, including the research strategy, data collection methods, and analytic procedures, is outlined in Section 3. The results of the analysis are presented in Section 4, with an emphasis on the vulnerabilities highlighted throughout the IoT architectural layers. The findings' consequences are covered in Section 5, while Section 6 presents the conclusion and suggestions for future research.

2. Background of the Research

Security and privacy are among the most important aspects of any system, and in healthcare, their importance cannot be overemphasized [30]. Clinical information, and in particular patient healthcare data, is considered sensitive information and mostly private as it may contain patient private information such as medical background, health conditions, and emergency contact details. This information becomes even more sensitive when it contains confidential medical data on sexual and physiological health or disease and drug misuse that patients generally refuse to share with anyone but their doctors. Moreover, over time patients' medical records become a rich repository of all types of information, including medical history, dietary and genetic data, and even family medical history.

Despite the significant IoT potential, security and privacy issues have limited the widespread deployment of IoT technologies in the healthcare domain [36]. Currently, healthcare information and data are exposed to more types of security and privacy breaches compared to the last 20 years, as reported by the Health Insurance Portability and Accountability Act (HIPAA) [37]. HIPAA is a US federal law that provides standards to protect sensitive patient health information from being disclosed without the patient's consent or knowledge [38].

The main issue of security and privacy breaches in healthcare is the lack of proper synchronization of security and privacy issues and technologies in the particular domain of healthcare [39]. This issue becomes further accentuated considering the new technologies such as mobile devices, cloud services, and remote applications being deployed in the healthcare system [40,41]. While these technologies have improved data accessibility in healthcare, they have also increased the risk of security and privacy breaches of health information.

The increasing dependence on technologies may result in increasing the vulnerability of population health data and makes citizens' healthcare information susceptible to different types of threats and misuse that may, in turn, have devastating impacts on both patients and healthcare organizations [42,43]. The lack of consideration for security and privacy in healthcare and IoT technologies is also a barrier to fully exploiting the potential of these technologies to address current challenges in healthcare. Therefore, it is crucial to clearly define and understand the concepts of security and privacy in healthcare. For this study, the following definitions were adopted from [44]:

Security: Protective measures that enable a healthcare organization to perform its critical functions despite risks posed by threats to its use of information systems.

Privacy: The restriction of access to healthcare information in accordance with laws and policies.

The significance of maintaining security and privacy in healthcare cannot be overstated, as evidence shows that security breaches in medical data can result in financial losses, medical identity theft, and discrimination [45,46]. Even the inappropriate disclosure of healthcare data can disrupt patients' privacy and expose them to further problems [47]. In some cases, people's health credentials can be stolen during medical treatments, leading to economic losses for healthcare providers and potential health consequences for patients due to the manipulation of their medical records [48].

Statistics show that breaches in healthcare records create financial concerns, psychological harm, medical identification fraud, and potential social stigma or prejudice against people with a specific condition [49]. Victims' health credentials are hijacked in some situations for medical treatment, which causes financial losses for healthcare institutions and could also have health effects for individuals owing to the alteration of their medical files [49]. Even inadvertently leaking healthcare data can affect patients' privacy [50,51] and expose them to additional issues like those described above.

Table 1 presents the number of healthcare data breaches from 2009 to 2022. Over the period of 2009 to 2022, the Health and Human Services (HHS) Office for Civil Rights has received reports of 5150 large-scale healthcare data breaches, affecting a total of 382 million healthcare records, equivalent to over 1.2 times the population of the United States. In 2018, one healthcare data breach of 500 or more records was reported daily. But five years later, the situation has worsened, with an average of 1.94 daily reports of such breaches in 2022.

Table 1. Healthcare Data Breaches Based on [37,52] Statistics.

Year	Number of Breaches (500+)	Healthcare Data Breaches (in Millions)
2022	707	50,000,000
2021	714	42,000,000
2020	642	34,000,000
2019	512	38,377,277
2018	368	12,069,868
2017	358	5,127,646
2016	329	16,657,540
2015	270	114,306,777
2014	307	12,901,859
2013	274	33,335,000
2012	209	23,335,000
2011	196	5,011,100
2010	198	1,151,000

To enhance security and address growing concerns over medical data privacy, various organizations have developed and established standards, policies, guidelines, protocols, and regulations for healthcare technology vendors and service providers. These regulations aim to protect healthcare systems and resources from intentional and unintentional risks while providing efficient and cost-effective clinical services to citizens [53]. Organizations such as HIPAA, the State Alliance for eHealth, and the National Governors Association Centre for Best Practices have provided rules and guidelines for using technology in healthcare that are widely recognized globally. Despite the potential benefits, IoT technologies offer to the healthcare system, the rapid advancements in technology have increased the vulnerability of population health data and exposed patients' information to various threats and misuses [54].

IoT Risks and Vulnerabilities in Healthcare

The use of IoT technology in the healthcare industry continues to grow. In 2022, the worldwide market for IoT in healthcare was estimated to be worth USD 99.58 billion. This market is predicted to increase to USD 486.34 billion by 2031, with a compound annual growth rate of 19.27% over the forecast period of 2023 to 2031 [55].

Studies have shown that the implementation of IoT in healthcare can improve patient outcomes and experience, increase the efficiency and productivity of healthcare organizations, and reduce costs [11,56]. For instance, wearable devices such as smartwatches and continuous monitoring systems can provide real-time monitoring of vital signs and early detection of potential health problems [57]. Therefore, this creates great opportunities for medical experts and the healthcare industry. However, security and privacy remain significant challenges in the implementation of IoT in healthcare. Health organizations reported security concerns as the main barrier to IoT adoption [58], while data privacy

and data protection ranked as the second largest concern among healthcare organizations [47]. Babar, et al. [59] claimed that having every 'thing' connected, new security and privacy problems arise, e.g., confidentiality, authenticity, and integrity of data sensed and exchanged by 'things'.

As most IoT devices and their applications are wireless in nature, security and privacy concerns are among the top areas of concern. A more highlighted issue with IoT in healthcare is the security of data transmitted from a patient's device through wireless media, which exposes the patient's data. In addition, this problem with IoT devices has been escalated due to the diversity of resources and the network heterogeneity [5]. The violations or privacy concerns discussed may restrict users from taking advantage of the full benefits IoT brings to the healthcare system [60].

Since IoT is a combination of heterogeneous networks of sensor network, mobile communication network, and the Internet, the same security and privacy threats of each of these networks directly pose security and privacy problems for IoT [5,61]. In addition, privacy protection problems, heterogeneous network authentication, and access control problems, information storage and management [5], confidentiality, and trustworthiness are other security areas for further consideration and investigation.

To address these security and privacy concerns, healthcare organizations are encouraged to implement robust security measures such as encryption, secure data storage, and access controls [62]. Additionally, there is a need for healthcare organizations to prioritize the development of industry-wide security and privacy standards for IoT devices and systems [54].

In 2020, a comprehensive survey of IoT in healthcare reported that the healthcare industry is one of the most promising fields for IoT applications due to its ability to transform healthcare services, reduce costs, and improve patient outcomes [26]. The survey also highlighted the importance of addressing the security and privacy challenges that arise with IoT systems in healthcare. Similarly, a 2020 study found that IoT technology can improve healthcare delivery by enhancing the quality of services, reducing healthcare costs, and improving patient outcomes. However, the study also cautioned that the risks associated with the use of IoT technology in healthcare, including cybersecurity concerns, must be managed appropriately [34].

While previous studies have explored the security and privacy concerns of IoT in general, there is however a notable research gap regarding the layered aspect of the IoT structure in the healthcare setting [28–30,35]. Each of the IoT layers, which includes perception, network, and application, has its security risks that when ignored stand to expose patient-sensitive information and the integrity of the data at stake. For example, the perception layer which collects data through sensors and wearable devices within the IoT infrastructure is highly prone to both physical attacks and data theft. The network layer where the data are encapsulated and sent from one point to another is faced with threats like eavesdropping, DDoS attacks, and man-in-the-middle strategies. The application layer which is a user interface and data storage system is vulnerable to weak authentication security, malware, and ransomware intrusions [32,34,35]. Reviewed studies always negate giving such distinct levels an inclusive scrutiny and treat them separately by concentrating on their security measures from the network scope [30,31]. It is therefore very important and timely to conduct such reviews along the IoT system layers for all IoT healthcare systems and deficiencies addressed. Table 2 compares our survey with other survey papers in the same area to demonstrate the gap in the literature this paper is trying to fill.

Table 2. Comparison Table: Survey Papers on IoT Security and Privacy in Healthcare.

Focus Area	IoT Layers	Recent Insight (2017–2023)	Limitation	Reference Paper
IoT security risks and threats to the healthcare sector	Network and application layers	No	Lacks updated solutions and real-time healthcare device security	[63]
Security and privacy issues in modern healthcare systems	Perception, network, and application layers	No	Does not provide practical solutions for recent IoT developments	[64]
Security and privacy in healthcare 4.0	Network, application, and cloud layers	No	Lacks focus on real-time healthcare IoT devices (perception layer)	[29]
IoMT edge network security issues	-	No	Lack of focus on IoT layers and structure	[65]
IoT security and privacy in healthcare-specific contexts	Perception, network, and application layers	updated to 2023		Our paper

Several studies have identified the security and privacy issues and attacks facing healthcare IoT systems. For example, a 2021 review of the existing literature on privacy and security in IoT-based healthcare systems identified the key challenges and potential solutions [27]. Another 2019 study reviewed the existing literature on IoT technologies and applications in healthcare and presented a case study of an IoT-based healthcare system [35]. Additionally, a 2018 study identified the security and privacy challenges in IoT-based healthcare systems and proposed potential solutions for addressing these challenges [66]. Some of the key security and privacy challenges identified in these studies include the vulnerability of IoT devices to attacks, data breaches, and unauthorized access to sensitive patient information.

Owing to the highlighted severity of the security and privacy concerns, a comprehensive review of the existing research studies is essential to provide a clear understanding of their types and prevalence. This review aims to identify the security and privacy issues and attacks facing healthcare IoT systems and how they manifest at the perception layer, the network layer, and the application layer. By doing so, this review will provide a foundation for future research and practical solutions to address the security and privacy challenges of IoT-based healthcare systems.

3. Research Methods

This research attempts to answer the following research question:

RQ: What are the security and privacy issues of each architectural layer of IoT technologies (i.e., the perception layer, the network layer, and the application layer) concerning their application in healthcare?

Our aim was to explore different types of security and privacy problems in each architectural layer of IoT in healthcare applications so that by addressing them, technology vendors and service providers can be informed to take them into consideration for their future technology/service development. To achieve our objective, we conducted a semi-systematic literature review as recommended by [67] to provide an overview of the research area over time.

3.1. Search Strategy and Selection Criteria

The search strategy for this review will involve a comprehensive search of various electronic databases, including Google Scholar, PubMed, IEEE, Scopus, and Science Direct. All potential variations in the keywords searching phrases in all databases were included in search engines: "Internet of Things" OR "IoT" AND "Healthcare" OR "Health Care" AND "Security" AND "Privacy". The search strategy will be designed to identify all relevant studies published between 2017 and 2023, to gather more up-to-date background information on this field.

The selection criteria will be based on predefined inclusion and exclusion criteria. Inclusion criteria will include studies that discuss security and privacy issues and attacks in healthcare IoT systems. Exclusion criteria will include studies not meeting the inclusion criteria and studies published before 2017.

In the first phase, 279 studies were obtained from various databases and placed into the dataset generated using Endnote software. A total of 13 duplicate papers were found and removed during the initial screening of the paper titles. Then, after skimming the abstracts, 112 research papers were removed based on the exclusion criteria (Table 3).

Table 3. Inclusion and Exclusion Criteria.

Inclusion Criteria	Exclusion Criteria
Studies in the English language	Articles not in the English language
Studies that discussed IoT security issues in healthcare	Studies with a pure technical focus
Conference proceedings and journal articles	Book chapters, company reports, letters, thesis, and abstracts
Studies after 2017	Studies with medical and healthcare focus only
	Studies on smart homes with no discussion of healthcare

After scanning the full text of 154 articles and applying the specified exclusion criteria, 85 articles were excluded. After this, 69 articles met the requirements for inclusion and were considered for a detailed search and analysis. Figure 2 illustrates the overall procedure and details of the research paper selection procedure.

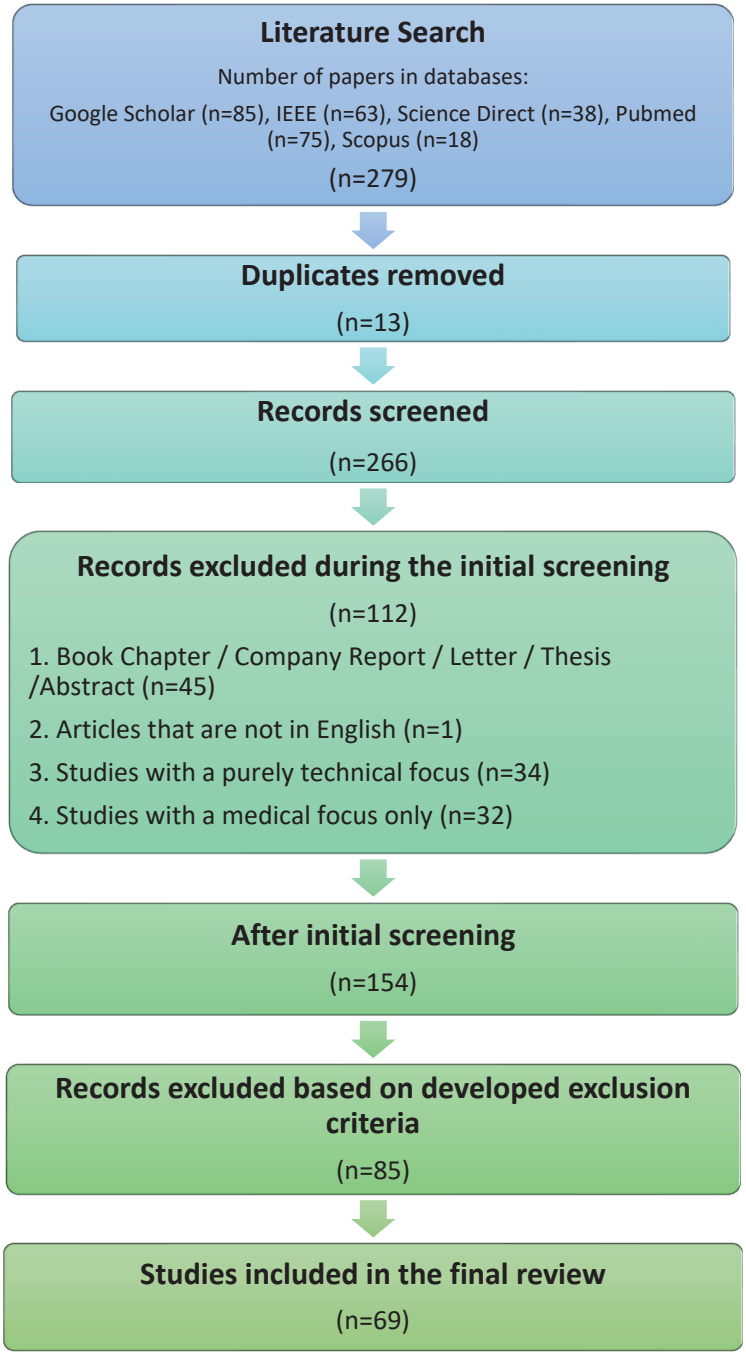


Figure 2. Literature review procedure.

3.2. Data Extraction and Analysis

The data extraction will involve systematically extracting relevant data from the selected studies using a predefined data extraction form. The extracted data will include study characteristics, such as the author, year of publication, study design, sample size, and study objectives. Additionally, the data extracted will include details about the security and privacy issues and attacks, such as the type of issue or attack, its prevalence, severity, and how it manifests at the perception, network, or application layers.

The analysis of the extracted data will involve a thematic analysis approach, where the data will be categorized into themes and sub-themes based on the research question and objectives. The analysis will be conducted using the appropriate software tool, NVivo, to manage and organize the data.

This phase collected data from selected articles and assigned them to predetermined groups. Several additional categories appeared at this step. Deductive and inductive coding is used in combination with this research method. Inductive coding is a non-predefined exploratory study, while deductive coding uses classifications as a guideline for the coding stage [68].

The pilot test outlined three essential security and privacy areas: the perception layer, the network layer, and the application layer of IoT devices in healthcare, used as a predeveloped dataset for deductive coding. In addition, new application areas were identified and introduced as new codes during the coding process. The combined coding method assisted in the coding phase using deductive and inductive coding to analyze the developing codes.

NVivo software (version 14) was used to organize the coding and data processing for data analysis. The review aimed to gather qualitative data for the investigation to comprehensively review existing and potential IoT security and privacy problems in healthcare. Due to the large amount of data, review analysis can be challenging. However, the process is manageable since a methodical approach is taken: Simply capture the main conclusions of all included articles by study type, and then evaluate the collection refers to results based on the frequency of data, to create a summary of results from multiple qualitative studies.

4. Results

The research results are depicted in both visual and narrative forms. The results of the data analysis from the records are included and showcased any correlations and patterns found in the collected data, which can help improve comprehension of the study outcomes and aid in addressing the research question.

IoT security and privacy threats can be categorized as physical, information and management [5], or system and information [60]. These threats can be passive or active. In the passive risks, attackers can capture information without impacting network behavior, while in the active ones, intruders can prevent or slow down administering the service [69]. As suggested by [5], IoT vulnerabilities can fall into the three layers of perception, network, and application (Figure 3).

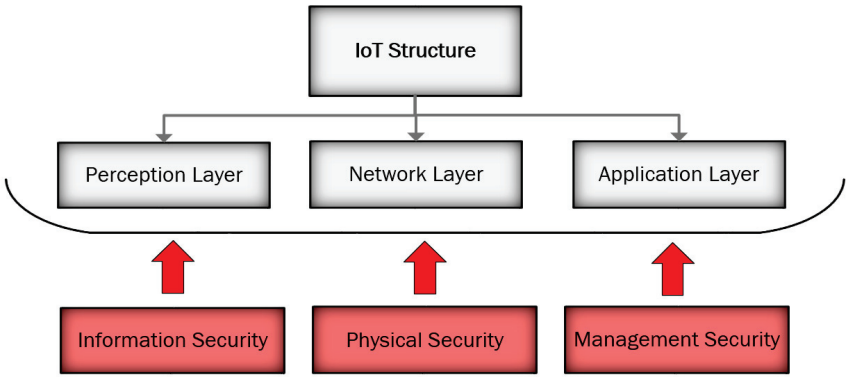


Figure 3. IoT structure and possible security threats.

4.1. Overview of IoT Security and Privacy in Healthcare

The integration of IoT in healthcare has the potential to benefit medical professionals and the healthcare industry greatly, but security remains a major barrier to its widespread adoption and implementation [58]. The primary concern is securing the data transmitted from patients' devices through wireless networks, which may result in unencrypted patient data being revealed [70]. The lack of security and privacy, especially with many IoT devices and applications being wireless, raises concerns among users regarding the full utilization of IoT capabilities in the healthcare system [43,71].

Networks and media are vulnerable to security and privacy threats, particularly in the healthcare industry, where personal and confidential information is involved [72]. The diversity of IoT devices and the heterogeneity of networks only exacerbate these security and privacy concerns [73]. To address these challenges, it is crucial to consider heterogeneous network authentication, privacy protection, access control, information management, confidentiality, and trustworthiness in the implementation of IoT in healthcare [72].

Users hesitate to adopt IoT-based healthcare services due to security and privacy concerns. Nevertheless, addressing these challenges is crucial for enhancing technology acceptance and reducing psychological and social barriers to IoT usage in healthcare [74].

Incorporating IoT in healthcare is not just about technological evolution but also involves the complex interplay of data privacy, security, and ethical considerations. The pervasive nature [75] IoT devices allow them to continuously gather and transmit sensitive health data, thus raising major concerns regarding data confidentiality and integrity. The nature of complexity mandates a multidisciplinary approach to security involving technical, ethical, and even legal expertise.

The threat landscape for IoT in healthcare has shown to be dynamic and multifaceted. IoT devices, by their very nature of always being interconnected and interdependent, create a vast attack surface [76] for cybercriminals. The risks are not just about data breaches but also the risk of manipulating the devices' functionalities, which creates grave security consequences for the health and safety of patients. The security of IoT in healthcare goes beyond the traditional cybersecurity challenges, thus requiring innovative solutions that address both data security and the physical safety of patients.

Enhancing security and privacy in IoT-based healthcare services can increase trust and adoption, thereby improving the overall healthcare system. Researchers have been exploring various security and privacy challenges in IoT applications in healthcare, and resolving these issues can guide technology suppliers and service providers to meet the requirements for potential advanced technology growth.

In this study, the design of IoT was used to investigate various issues. As a result, the vulnerabilities in IoT in healthcare were categorized into three layers, namely the perception, network, and application layers. The following subsections go over each of these categories.

4.1.1. The Perception Layer

The IoT perception layer is a vital element of healthcare IoT systems, enabling connected devices and sensors to capture and transmit data for analysis and decision-making purposes. Since IoT is an integration of multiple different networks, there are compatibility issues [61] that prone the overall IoT to various security risks. The security issues of this layer include those dealing with physical devices as well as data collection. The proliferation of these devices and sensors has led to significant security and privacy risks, as a variety of IoT devices [61,77], make implementing a single and unified security design burdensome. For example, there are two RFID-specific security vulnerabilities in this layer: uniform coding and conflict collision [61]. The variation in IoT devices may also cause a problem in data aggregation, which, in turn, jeopardizes data integration [61]. This issue is counted as one of the most critical security issues in healthcare [78].

As healthcare IoT systems continue to evolve and expand, it is vital to remain vigilant in addressing the security and privacy risks associated with the IoT perception layer. Collaboration among healthcare professionals, cybersecurity experts, and technology vendors is essential to mitigate these risks and ensure the safety and integrity of patient data and healthcare systems. It is crucial for healthcare organizations to implement security measures and stay informed of the latest security threats to ensure the safety of their patients and their systems.

Various types of security and privacy issues have been reported at the perception layer of healthcare IoT systems, including unauthorized access, data breaches, identity theft,

data manipulation, and denial-of-service (DoS) attacks [79]. These attacks can have serious consequences, including compromised patient data, system downtime, and financial loss.

One of the significant challenges of the IoT perception layer is the vulnerabilities that come with it. Several studies have investigated security and privacy issues and attacks at the IoT perception layer in healthcare IoT systems. A sniffing attack is a common example when an attacker intercepts data as they are transmitted between connected devices, allowing them to read and potentially modify the data. A replay attack involves the attacker intercepting and then resending a message, potentially leading to unintended consequences. A man-in-the-middle attack occurs when an attacker intercepts data between two parties and alters the data, leading to the parties unknowingly communicating with the attacker rather than each other. Finally, a buffer overflow attack occurs when an attacker sends more data than a system's memory can handle, potentially leading to a system crash or other unintended consequences [64].

One study found that some IoT medical devices have inadequate security controls in place, leaving them vulnerable to attack [80]. Therefore, in healthcare applications, probably the most crucial tasks to provide security are making IoT devices resistant to physical/device tampering and preventing physical/device manipulation, as they are the most common attack in this layer [81]. Since many IoT devices may be placed without extra security, attendance, or involvement in patients' environments, they are still subject to severe attacks. This also makes them susceptible to physical tampering, which could allow intruders to query the IoT devices and search the collection devices to exploit and steal data from patients [81]. Therefore, for the IoT healthcare system to maintain security, node authentication is the essential first step in identifying any malicious nodes because of physical attacks. Physical or device tampering can be serious in healthcare since IoT devices in healthcare are designed and programmed to capture, store, and protect patients' healthcare information.

Node authentication is the first critical step in identifying malicious nodes due to physical attacks for the IoT of healthcare systems to keep their safety. Physical or device exploitation in healthcare can be significant because many devices in healthcare have been designed and developed to record, store, and protect the health records of individuals [17]. In addition, data encryption requires protected distribution. Flexible encryption algorithms are suitable in cryptography because of hardware limitations. The resources of the IoT devices (battery power, memory, and CPU) are not drained rapidly. Moreover, some IoT medical devices have limited resources, such as memory and processing power, making it challenging to implement effective security controls. IoT sensors mostly have little external security features or protection, so they cannot handle complicated security and public key encryption techniques/algorithms or frequency hopping communication [5,82]. Also, most IoT devices just support short passwords [61] that compromise trust management in devices. These types of deficiencies result from memory, computational, and energy constraints; therefore, they make these devices prone to physical tampering and jamming attacks [61,69,77].

Additionally, the lack of industry standards for IoT devices can lead to inconsistencies in security protocols and make it difficult to ensure interoperability. The list of security and privacy issues of medical IoT in the perception layer is provided in Table 4.

Another study found that healthcare organizations are particularly vulnerable to phishing attacks, which can be used to steal sensitive patient information [83]. Such findings highlight the need for robust and proactive measures to address the security and privacy risks associated with the IoT perception layer, although the main reason of compromising security and privacy of IoT in this layer is hardware limitations [77].

Furthermore, some inherent challenges make securing the IoT perception layer in healthcare more difficult. For instance, the large number of connected devices and sensors in a typical healthcare IoT system makes it difficult to manage and secure them all.

To address these challenges, ongoing research is needed to identify vulnerabilities and develop effective security solutions for the IoT perception layer in healthcare. For instance,

researchers have proposed using machine learning algorithms to identify potential attacks and anomalies in data patterns, allowing for rapid detection and response to security threats [84]. Other researchers have suggested using blockchain technology to create a tamper-proof audit trail for medical data, ensuring its integrity and confidentiality [85].

It is also essential for healthcare organizations to stay informed of the latest security threats and implement best practices for securing their IoT systems. This includes using strong passwords, regularly updating software and firmware, and encrypting sensitive data in transit and at rest. Regular security audits and risk assessments can also help identify vulnerabilities and weaknesses in the system, allowing for proactive security measures to be implemented.

In conclusion, the IoT perception layer plays a crucial role in healthcare IoT systems, but it also poses significant security and privacy risks. Healthcare organizations must proactively secure their IoT systems and stay informed of the latest security threats and best practices. Ongoing research and collaboration among healthcare professionals, cybersecurity experts, and technology vendors are also necessary to develop effective security solutions and ensure the safety and integrity of patient data and healthcare systems. With proper security measures in place, the IoT perception layer can continue to enable the development of innovative healthcare technologies that improve patient outcomes and enhance the quality of care.

Table 4. Security and Privacy Challenges of the IoT Perception Layer.

Issue or Attack	Explanation and Highlight	Countermeasure
Man in the Middle	An attacker intercepts and manipulates the communication between an IoT device and the network, potentially stealing or altering sensitive information [86,87]. In 2019, a report found that almost 25% of attacks targeted the healthcare industry [88].	Implementing encryption for all communications between IoT devices and the network [86], using secure protocols [89], and implementing certificate-based authentication [90].
Denial of Service (DoS)	An attacker makes an IoT device unavailable by overwhelming it with traffic or by exploiting a vulnerability in the device, potentially compromising the security and privacy of patient information [91]. According to [92], 162 attacks targeted healthcare businesses, affecting 900 clinics, hospitals, and organizations.	Rate-limiting and filtering mechanisms [93] can be implemented to prevent traffic flooding. Network segmentation [54] can isolate IoT devices, and firewalls and intrusion detection systems [94], can prevent unauthorized access.
Distributed Denial of Service	An attacker floods an IoT device with a large amount of traffic, causing it to become unavailable and potentially leading to a failure of the device or the entire system [95,96]. In 2016, many IoT nodes were used in the DDoS attack known as "Mirai" [97].	Implementing a firewall [93], using intrusion detection and prevention systems [94], and using traffic filtering techniques [96,98] can help prevent DDoS attacks.
Malware and Virus	Malicious software infects an IoT device, potentially compromising the security and privacy of patient information [99].	Regularly updating and patching software [54], using antivirus and anti-malware software [100], and restricting the use of untrusted devices on the network [101].

Table 4. *Cont.*

Issue or Attack	Explanation and Highlight	Countermeasure
Phishing	This occurs when an attacker tricks individuals into revealing sensitive information or downloading malware onto an IoT device, potentially compromising the security and privacy of patient information [91].	Providing training to employees on how to recognize phishing attempts [102], implementing email filtering systems [13], and using multifactor authentication [101].
SQL Injection	This occurs when an attacker injects malicious code into a database, potentially compromising the security and privacy of patient information [103].	Implementing input validation [104] and regularly updating and patching database software [54].
Cross-Site Scripting	This occurs when an attacker injects malicious code into a database, potentially compromising the security and privacy of patient information [103].	Implementing input validation [104] and regularly updating and patching database software [54].
Insider Threats	An employee or contractor with access to sensitive information engages in malicious behavior, potentially compromising the security and privacy of patient information [105].	Employees and contractors should be trained on the importance of security and privacy in the healthcare sector [106]. Strict access control mechanisms, such as role-based access control [107].
Ransomware	An attacker encrypts sensitive patient information and demands a ransom in exchange for the decryption key, potentially compromising the security and privacy of patient information [74].	Regular backups of sensitive data can prevent data loss in the event of a ransomware attack [102]. Employees should also be trained on how to recognize and avoid phishing attacks, which are commonly used to deliver ransomware [102].
Jamming	This occurs when an attacker interferes with the communication between an IoT device and the network, potentially causing the device to fail and leading to security and privacy incidents [108]. In 2020, a hospital in Europe reported that its communication systems had been disrupted by a jamming attack, compromising patient information and care [53].	Encryption should be employed for communication between IoT devices and the network [109]. Secure communication protocols should be used.
Reply Attack	An attacker intercepts and reuses valid data transmission, potentially compromising the security and privacy of patient information [110].	Secure communication protocols that include message authentication codes [110], should be utilized. Encryption should also be employed for communication between IoT devices and the network.

Table 4. *Cont.*

Issue or Attack	Explanation and Highlight	Countermeasure
Side-Channel	This occurs when an attacker uses information obtained from the physical operation of an IoT device to exploit security vulnerabilities, potentially compromising the security and privacy of patient information [92]. In 2020, researchers disclosed a side-channel attack on a popular medical device, showing how an attacker could use information obtained from the physical operation of the device to compromise patient information [111].	Secure coding practices, such as constant-time coding [54], should be employed.
Eavesdropping	An attacker intercepts and listens to the communication between an IoT device and the network, potentially stealing sensitive information [87].	Encryption should be used for communication between IoT devices and the network [54]. Secure authentication methods like certificates or tokens can also be used to protect against eavesdropping [90].
Tampering Attacks	This occurs when an attacker modifies the data or configuration of an IoT device, potentially compromising the security and privacy of patient information [87].	Digital signatures can be used to verify the integrity of data [112]. Secure software update mechanisms [113] can also prevent tampering.
Insufficient Cryptographic Algorithms	The insufficient protection of sensitive data transmitted between IoT devices and networks due to the use of weak encryption protocols or lack of encryption altogether [114]. According to a report in 2019, almost 60% of all IoT devices in healthcare organizations were using outdated encryption technologies, posing a risk to patient privacy [92].	Secure encryption protocols should be used for communication between IoT devices and the network.

4.1.2. The Network Layer

The network layer, responsible for data transmission between devices, presents unique security challenges in healthcare IoT systems. The very flexibility and scalability of IoT networks [115] along with the mobility of devices complicate securing this layer [77].

The vast array of IoT devices often utilizes diverse network protocols for data transmission [92] This heterogeneity makes it challenging to implement a single, unified security protocol that caters to all network types (wired and wireless). Even existing security models have inherent flaws [77].

Establishing robust security standards across various networks, including cellular (4/5G), ad hoc, and Wi-Fi networks, is crucial [36]. It is important to note that security vulnerabilities in this layer can resemble conventional network security challenges. The network layer is susceptible to attacks like Trojan horses, viruses, spam, packet spoofing, route falsification, and flooding attacks, potentially leading to information disclosure [116].

The widespread use of wireless networks in healthcare IoT makes data inherently more vulnerable to eavesdropping due to the open nature of wireless communication [117].

Another significant security concern lies in the use of insecure communication protocols. Many healthcare IoT devices rely on unencrypted protocols like HTTP, which attackers can exploit to compromise patient data confidentiality and integrity [118]. Unencrypted data transmissions expose information to interception, modification, or deletion during transfer.

The network layer is also susceptible to various threats such as packet spoofing, route falsification, and flooding attacks [60]. A prevalent issue is the lack of encryption within the network layer [119]. Encryption is a crucial security measure that helps protect patient data as they are transmitted over the network [120]. Without encryption, sensitive patient data can be intercepted and read by cybercriminals, compromising patient privacy and confidentiality [114]. Encryption provides an additional layer of security that makes it more difficult for attackers to access and read the transmitted data.

Healthcare IoT devices are also susceptible to distributed denial of service (DDoS) attacks, which can overwhelm network resources and cause downtime for critical medical systems [121]. DDoS attacks can result in service disruptions, and in healthcare, such disruptions can be life-threatening. According to [36], DoS attacks can be a crucial problem for multidomain infrastructure and, therefore, for IoT medical applications. Healthcare IoT systems must have appropriate security controls in place to detect and mitigate DDoS attacks promptly.

Legacy devices and outdated software pose a significant security risk to healthcare IoT systems, as they often lack security patches and updates [54]. These devices and software can have known vulnerabilities that attackers can exploit to gain unauthorized access to sensitive patient data. Healthcare organizations must take appropriate measures to ensure that legacy devices and software are updated and protected against known vulnerabilities.

The use of mobile devices by healthcare providers to access IoT systems outside of the secure hospital network creates a significant security risk. Mobile devices can be infected with malware, targeted by phishing attacks, and can be vulnerable to network eavesdropping [122]. Healthcare organizations should ensure that mobile devices used to access IoT systems are appropriately secured and use secure communication protocols to communicate with IoT devices.

Several studies have identified vulnerabilities in the network layer of healthcare IoT systems that cybercriminals can exploit to launch various attacks. For example, a study by Obaidat et al. [101] identified vulnerabilities in the network layer that could be exploited to launch attacks, such as replay attacks, man-in-the-middle attacks, and message injection attacks. Similarly, another study by Islam et al. [123] reported on vulnerabilities in the ZigBee communication protocol, which is widely used in healthcare IoT systems. The researchers found that the lack of encryption and authentication mechanisms in ZigBee networks made them vulnerable to a range of attacks, including traffic sniffing, DoS, and spoofing attacks.

Some of the security and privacy issues of medical IoT in the network layer are presented and defined in Table 5.

To mitigate these risks, healthcare organizations must implement appropriate security controls such as encryption, authentication, and access control mechanisms [124]. Healthcare organizations should also use up-to-date software and security patches, conduct regular security assessments and testing, and ensure that legacy devices and software are updated and protected against known vulnerabilities. By remaining vigilant and proactive, healthcare organizations can prevent cyberattacks and safeguard the privacy and confidentiality of patient data.

Table 5. Security and Privacy Challenges of the IoT Network Layer.

Issue or Attack	Explanation and Highlight	Countermeasure
DOS Attacks	This type of attack involves overwhelming a network with a large amount of traffic, which can result in the disruption of critical services, such as electronic medical records systems or medical device networks [121].	Implementing network security solutions such as firewalls, intrusion prevention systems (IPS), and network behavior analysis systems to detect and block DDoS attacks [86,125].
IP Spoofing	This type of attack involves forging the source IP address of a network packet, which can be used to launch attacks or disrupt services, such as remote monitoring systems or telemedicine platforms [126].	Using strong authentication and encryption techniques, such as IPsec, to secure data transmission and prevent IP spoofing [127,128].
ARP Spoofing	This type of attack involves forging ARP (Address Resolution Protocol) messages on a network, which can be used to redirect network traffic and launch man-in-the-middle attacks, such as intercepting sensitive information transmitted between medical devices and servers [129].	Implementing ARP spoofing detection tools, such as ARPwatch or ARP poisoning detection tools, to detect and prevent ARP spoofing attacks [104].
DHCP Spoofing	This type of attack involves forging DHCP (Dynamic Host Configuration Protocol) messages on a network, which can be used to assign unauthorized IP addresses to IoT devices and launch man-in-the-middle attacks, such as intercepting sensitive information transmitted between medical devices and servers [130,131].	Implementing DHCP spoofing protection measures, such as static IP assignment, to prevent unauthorized IP assignments [130,131].
Sniffing Jamming	This type of attack involves intercepting and analyzing network traffic, which can be used to collect sensitive information, such as medical records or patient data, or launch attacks [104].	Using encryption techniques, such as SSL/TLS, to secure data transmission and prevent sniffing attacks [104,132].
Packet Injection	This type of attack involves injecting malicious packets into a network, which can be used to launch attacks or disrupt services, such as electronic medical records systems or medical device networks [104].	Implementing network security solutions, such as firewalls, intrusion detection systems (IDS), and network behavior analysis systems, to detect and prevent packet injection attacks [104].
Malware Attacks	This type of attack involves infecting devices with malicious software that can steal sensitive information or disrupt device operation [120]. In 2019, the malware known as Ryuk was found to be infecting hospital systems, leading to widespread disruption and ransom demands (A.).	Regular software updates and patches can prevent malware attacks. Implementing antivirus software and firewalls and using encrypted communication protocols can also help in preventing malware attacks [86,133].

Table 5. *Cont.*

Issue or Attack	Explanation and Highlight	Countermeasure
Man in the Middle	This type of attack involves intercepting and modifying network traffic, which can be used to steal sensitive information, alter device operation, or launch attacks [116]. In 2020, the malicious actor group known as APT10 was found to be using man-in-the-middle attacks to target healthcare organizations and steal sensitive information [134].	Encrypting all network traffic and using secure protocols such as SSL/TLS can prevent man-in-the-middle attacks. Implementing strong authentication mechanisms can also help in preventing these attacks [104].
Session Hijacking	This type of attack involves intercepting and taking over a user's network session, which can be used to steal sensitive information or launch attacks.	Implementing secure session management, using encrypted communication protocols, and implementing strong authentication mechanisms can help prevent session hijacking [54,135].
Forgery Attack	The use of forged or manipulated data to compromise the integrity of healthcare services and patient safety. An exploit leads an Internet browser to perform an inappropriate consumer application activity. A successful cyberattack can be catastrophic for both the patient and the healthcare organization [92].	Using digital signatures, access controls, encryption, network segmentation, blockchain technology, validating input data, and using intrusion detection and prevention systems to detect and prevent unauthorized access and modification of data in real time [65,133].
Hello Flood	A type of distributed denial-of-service (DDoS) attack that floods the target network with a large number of "hello" messages, overwhelming the network and causing it to crash or become unavailable [43,136]. "Hello" signals are defined in various routing protocols, allowing nodes to communicate with their neighbors [43].	Technical measures such as network segmentation, rate limiting, intrusion prevention systems, continuous network monitoring, and using content delivery networks, as well as organizational measures such as keeping IoT devices up to date and planning for disaster recovery [136].
Black Hole	A type of DDoS attack that exploits vulnerabilities in IoT devices to drop or discard legitimate network traffic, causing network congestion and disruption of healthcare services [137]. The attacker creates a fake node that accepts network traffic by claiming to have the quickest route. Network traffic could be forwarded to the fake node, which is used as a proxy server or ignored [138].	Using intrusion detection and prevention systems, implementing access controls, using network traffic analysis, rate limiting, keeping IoT devices up to date, deploying firewalls, and implementing network segmentation to isolate critical healthcare systems and data from less critical systems [137,139].

Table 5. *Cont.*

Issue or Attack	Explanation and Highlight	Countermeasure
Gray Hole	A type of attack where an IoT device selectively drops or modifies some network traffic, leading to degradation in service quality and potential data loss. This attack is similar to a black hole attack; instead of destroying all signals, it just removes a selection of those [140].	Using encryption, implementing access controls, network traffic analysis, intrusion detection and prevention systems, keeping IoT devices up to date, deploying firewalls, and implementing data redundancy [137,141].
Worm Hole	A type of attack where an attacker captures packets from one part of the network and tunnels them to another part of the network, bypassing network security measures [137]. By managing at least two network nodes or adding additional false nodes to the network, the attacker establishes a link between different parts of the network. The hacker gathers data from one point and replays it to the other after confirming the link [142].	Implementing secure communication protocols, using location-based authentication, network traffic analysis, intrusion detection, and prevention systems, implementing access controls, using digital signatures, and deploying firewalls [137].
Congestion Attack	A type of attack where an attacker floods the network with traffic, causing it to become congested and resulting in the degradation of service quality or even network failure [124]. This attack succeeds by flooding networks with unprocessed responses until their period runs out. Then, the lane is restricted to transactions whenever the highest number of unsolved requests (HTLCs) is achieved [143].	Using traffic shaping, implementing Quality of Service (QoS) policies, using rate limiting, implementing access controls, deploying intrusion detection and prevention systems, using network traffic analysis, and keeping IoT devices up to date with the latest security patches and firmware updates [65].

4.1.3. The Application Layer

This layer is responsible for data processing, storage, and presentation. It provides the interface through which users interact with IoT devices. As a result, healthcare IoT systems are vulnerable to a wide range of security and privacy issues and attacks at the application layer. The high number of IoT applications and their diversity and complexity could be considered the main cause of the security issues of the IoT caused by system integration [68,91].

One of the most common issues is the lack of data integrity and authenticity, which can lead to the spread of false information and negatively affect patient care [91]. Ensuring data integrity and authenticity can be achieved through implementing secure communication protocols, such as Transport Layer Security and Secure Sockets Layer to protect the data while in transit.

Another common issue is the use of default or weak credentials, which can easily be guessed or brute-forced by attackers, compromising patient privacy and safety [144]. To prevent such attacks, the healthcare industry should establish strong password policies, use multifactor authentication, and ensure that IoT devices are configured with strong passwords [145].

Moreover, several reports of attacks on the application layer in healthcare IoT systems include ransomware attacks that can disrupt critical medical services [54,146] and DDoS attacks [147] that can make the system unavailable. Such attacks can be prevented by implementing network security controls, such as firewalls and intrusion detection systems, to detect and mitigate attacks before they cause damage [148].

Malware attacks on healthcare IoT systems are also a concern. Malware can infect healthcare IoT devices, allowing attackers to steal sensitive patient data, disrupt device operations, and compromise patient safety [124]. To prevent malware attacks, healthcare organizations should ensure that all IoT devices have up-to-date security patches and conduct vulnerability assessments and penetration testing regularly [149].

Another challenge is that healthcare IoT systems often rely on third-party software and services [150], which can introduce additional security risks. Many of these software and services may have vulnerabilities or may not be configured securely, which can allow attackers to exploit them and gain unauthorized access to the system. Healthcare organizations should perform a thorough security assessment of all third-party software and services before integrating them into their IoT systems. Table 6 presents a variety of security issues in the application layer.

Table 6. Security and Privacy Challenges of the IoT Application Layer.

Issue or Attack	Explanation and Highlight	Countermeasure
DDOS Attacks	It can be targeted at medical systems, disrupting access to critical medical information and services [147].	Healthcare organizations can implement anti-DDoS measures such as traffic filtering, rate limiting, and network traffic monitoring. They can also consider using cloud-based anti-DDoS services, which can provide additional protection against DDoS attacks [147,148].
Insider Threats	Employees or contractors in clinical settings can intentionally or accidentally cause a data breach or steal information [86,151].	To mitigate the risk of insider threats, healthcare organizations can implement access controls and monitor user activity [151].
Remote Access Vulnerabilities	Remote access to medical systems can introduce security vulnerabilities if the access is not properly secured [54]. In 2019, a healthcare organization's remote access system was hacked, exposing the PHI of over 2 million patients [152].	To secure remote access to medical systems, healthcare organizations can implement secure authentication methods, such as multifactor authentication, and regularly update remote access software and systems [54].
Outdated Software and Systems	Outdated software and systems can be vulnerable to cyber-attacks, making it important for healthcare organizations to update their technology [54] regularly. In 2020, a healthcare organization's outdated software was found to have a vulnerability that allowed hackers to access sensitive medical information [153].	To mitigate the risk of outdated software and systems, healthcare organizations should regularly review and update their technology, including software, hardware, and systems [54].

Table 6. *Cont.*

Issue or Attack	Explanation and Highlight	Countermeasure
Malicious Apps	Malicious apps can be designed to steal information or launch attacks and can be easily installed on IoT devices [104,124].	To prevent the installation of malicious apps, healthcare organizations can implement strict app-vetting processes and regularly update mobile device management software [104].
Insecure Cloud Storage	If medical information is stored in the cloud, it can be vulnerable to attacks if the cloud storage is not properly secured [104]. In 2020, a healthcare organization's cloud storage system was hacked, exposing the PHI of over 1 million patients [37].	To secure cloud storage, healthcare organizations can encrypt sensitive information, regularly monitor access to cloud storage, and implement strict access controls [104]. They can also consider working with a trusted cloud service provider that specializes in secure cloud storage for the healthcare industry.
Physical Tampering	Physical tampering with IoT devices, such as removing or replacing components, can allow attackers to access sensitive information or launch attacks [54]. In 2021, a report by the U.S. Government Accountability Office found that medical devices such as pacemakers, insulin pumps, and CT scans were vulnerable to hacking due to physical tampering [154].	Healthcare organizations can implement strict access controls and regularly monitor IoT devices for signs of tampering [54,104].
Data Privacy and Identity	IoT interconnects devices from various manufacturers demanding the use of several authentications. Integrating multiple techniques to protect data privacy and identity is challenging [54,116,144].	Using encryption and access controls, implementing secure communication protocols, minimizing data, conducting regular security audits, using strong authentication and passwords, implementing data anonymization, and establishing a privacy policy [54,86].

To mitigate these risks, healthcare organizations must implement appropriate security controls in the application layer, including data encryption, authentication, and access control mechanisms. Healthcare organizations should also use up-to-date software and security patches, conduct regular security assessments and testing, and ensure that default or weak credentials are not used in their IoT systems. Additionally, organizations should provide their employees with cybersecurity training to increase awareness of the importance of security and best practices in IoT security.

In summary, healthcare IoT systems present several security and privacy challenges, particularly at the application layer. To mitigate these risks, healthcare organizations must implement appropriate security controls, including data encryption, access control, and regular security assessments and testing. Emerging technologies such as blockchain and secure software development practices can provide additional layers of security. Additionally, healthcare organizations should ensure that patients are fully informed about the risks and benefits of using IoT devices and provide them with options for controlling the use of their data. By taking these steps, healthcare organizations can ensure the pri-

vacy and confidentiality of patient data and prevent cyberattacks that could compromise patient safety.

4.2. Summary of Key Security and Privacy Threats

The main security and privacy issues, along with potential effects and mitigation options, are summarized in Table 7 below to deliver a clear perspective of the numerous threats found throughout each layer of the Health IoT architecture.

Table 7. Summary of Key Security and Privacy Threats in Health IoT.

IoT Layer	Key Security and Privacy Threats	Impact on Healthcare	Proposed Solutions
Perception Layer	- Device tampering - Data breaches - Replay attacks	- Compromised patient data integrity - Unauthorized access	- Physical security controls - Secure node authentication protocols
Network Layer	- Denial of service (DoS) - Eavesdropping - IP Spoofing	- Disruption of critical healthcare services - Data interception	- Encryption of network communications - Firewalls
Application Layer	- Malware attacks - Ransomware - Weak credentials	- Loss of sensitive patient information - Service downtime	- Regular software updates - Access control mechanisms
General Issues	- Lack of encryption - Inadequate authentication mechanisms	- Vulnerabilities in patient privacy - Data breaches	- Multifactor authentication - Data encryption

This summary table presents a comprehensive resource for the several security issues covered in the previous section. An organized overview of potential vulnerabilities that healthcare organizations need to solve is given by classifying these risks according to the IoT perception, network, and application layers. The potential impacts of these threats on patient safety, data integrity, and system availability are also highlighted in the table, showing how important it is to put in place appropriate countermeasures.

The main findings are presented in the table, highlighting the key areas in which healthcare providers should focus on strengthening their security operations. Organizations can significantly reduce the risk of security breaches and safeguard critical patient data by using the recommended mitigation techniques, improving the general dependability and security of Health IoT systems.

4.3. Taxonomy of Analyzed Vulnerabilities

A taxonomy that offers a clear explanation for selecting particular vulnerabilities for research has been developed, in addition to a review of the significant security and privacy issues. This taxonomy classifies the vulnerabilities according to a number of important factors that affect the performance and security of health-related IoT systems. Among these characteristics are:

- *Impact on Data Integrity:* Issues that could bring incorrect diagnosis or treatment decisions by jeopardizing the availability and integrity of patient data.
- *Threat to Patient Safety:* Challenges include tampered medical equipment or malfunctioning life support systems that threaten the health and safety of patients.
- *System Availability:* Vulnerabilities that impact IoT system availability and cause delays or disruptions in service for critical healthcare operations.
- *Attack Type:* This refers to the type of attack, which can be divided into three categories: application-based, network-based, and physical.

- *Exposure Level:* A measure of an IoT system's vulnerability to outside attacks, which can be impacted by communication protocols, network architecture, and device mobility.

An overview of this taxonomy and the vulnerabilities that belong to each characteristic can be viewed in Table 8 below.

Table 8. Taxonomy of Characteristics and Corresponding Vulnerabilities.

Characteristic	Description	Vulnerabilities
Impact on Data Integrity	Ensures that patient data are accurate and unmodified during its collection, storage, and transmission by focusing a strong focus on its integrity	Data tampering, packet spoofing, man-in-the-middle attacks, malware, replay attacks
Threat to Patient Safety	Detects weaknesses that can directly compromise the health and safety of patients by damaging medical equipment or postponing treatments	Device tampering, ransomware, DoS attacks, physical attacks on medical devices
System Availability	Concerns about risks that could block access to medical data or healthcare services, causing delays in treatment or system failures	Denial-of-service (DoS), distributed DoS (DDoS), congestion attacks, black hole and gray hole attacks
Attack Nature	Classifies vulnerabilities into three categories: application layer based, network layer based, and physical layer based	Physical attacks, malware, phishing, jamming, network attacks, SQL injection, cross-site scripting
Exposure Level	Evaluates the level of system exposure while considering the heterogeneity of networks and devices and how vulnerable they are to outside attacks	Eavesdropping, unauthorized access, sniffing attacks, weak encryption, unpatched vulnerabilities

Healthcare providers can use this taxonomy to prioritize security measures according to the specific threats they face, in addition to deploying it to better understand how risks are determined. A more detailed approach to risk management can be implemented by examining vulnerabilities utilizing these classifications, addressing both short-term threats and long-term security needs.

5. Discussion

In healthcare IoT systems, there are multiple layers of the system vulnerable to security and privacy risks. The architecture of a typical IoT platform in the healthcare domain and the associated threats are illustrated in Figure 4.

The perception layer is essential for data capture and transmission in healthcare IoT systems. However, as shown in Figure 4, the widespread use of devices and sensors in this layer has increased security risks, including unauthorized access, data breaches, and DoS attacks. This includes wearable and wellness devices, implantable medical devices (e.g., insulin pumps), medical equipment, and patient monitoring systems. Securing the perception layer is difficult due to the number of connected devices, limited resources, and lack of industry standards. Continuous research and collaboration among healthcare professionals, cybersecurity experts, and technology vendors are necessary to identify vulnerabilities and develop effective security solutions.

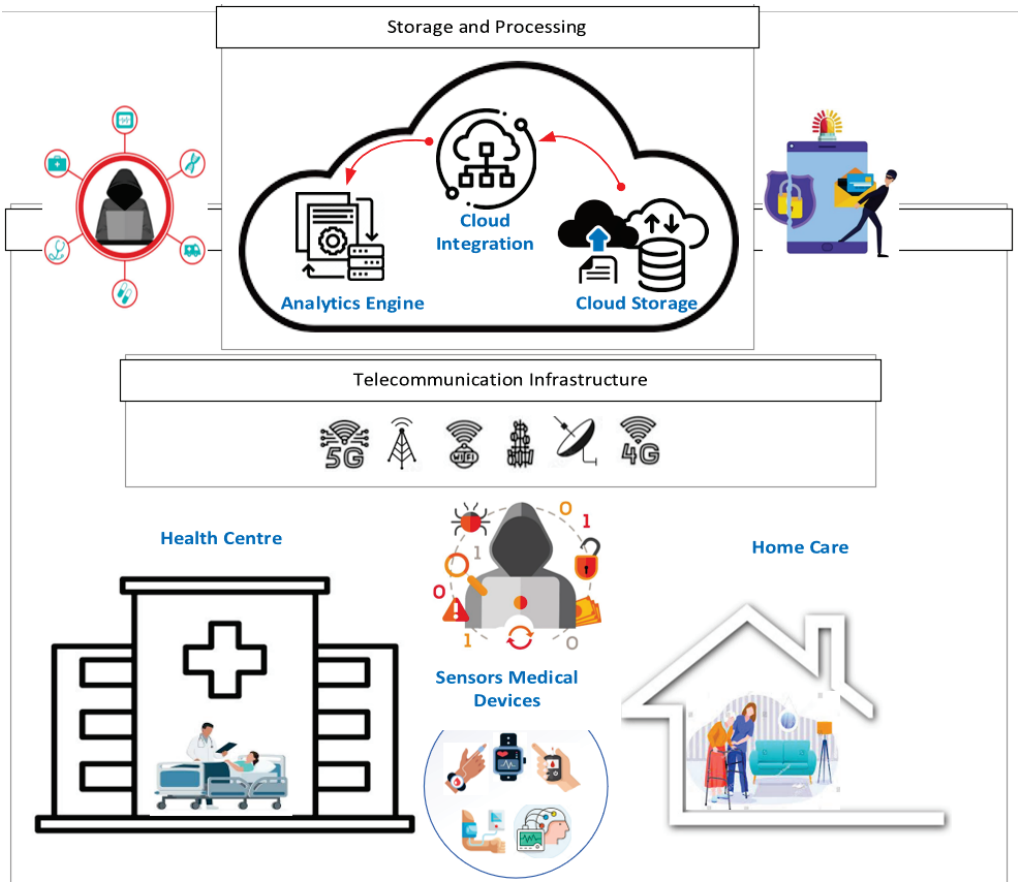


Figure 4. Typical IoT platform in healthcare and associated threats.

The network layer is also vulnerable to cyberattacks due to insecure communication protocols, inadequate authentication mechanisms, and the use of legacy devices and outdated software. Healthcare organizations need to implement appropriate security controls, such as using up-to-date software and security patches, conducting regular security assessments and testing, and ensuring that legacy devices and software are updated and protected against known vulnerabilities.

The application layer of healthcare IoT systems includes both data storage and processing and user interface and services. This layer is vulnerable to various security and privacy issues, including weak credentials, ransomware, and DDoS attacks. Healthcare organizations can mitigate these risks by implementing appropriate security controls, such as data encryption, access control, and regular security assessments and testing. Emerging technologies like blockchain and secure software development practices can provide additional layers of security. Healthcare organizations need to ensure that patients are informed about the risks and benefits of using IoT devices and provided with options for controlling the use of their data. Regular training and awareness programs can also help prevent security incidents and ensure that employees are aware of the risks associated with IoT devices.

Understanding how security threats correspond to each architectural component is crucial for implementing effective security measures. Based on Figure 4, Table 9 presents this mapping.

Table 9. Mapping of Security Threats to Architectural Components.

Architectural Component	Associated Threats	Potential Impact
Medical Devices and Sensors	Physical tampering Node capture Battery drainage attacks Sensor data manipulation	Compromised patient data False medical readings Device malfunction
Communication Infrastructure	Man-in-the-middle attacks DDoS attacks Protocol vulnerabilities Network eavesdropping	Data interception Service disruption Unauthorized access
Data Processing and Storage	Data breaches Malware infections Unauthorized access SQL injection	Privacy violations Data loss System compromise
User Interface and Services	Authentication bypass Session hijacking Cross-site scripting Phishing attacks	Unauthorized access Identity theft Service disruption

Standards and Solutions

The incorporation of IoT in healthcare mandates a strategic approach that goes beyond what our current conventional security measures offer. This involves developing a security-first approach within every healthcare organization, where the importance of data security and patient privacy is ingrained in every aspect of healthcare delivery. In other words, it should be a security [155] and privacy-by-design approach. Practitioners must gain a robust understanding of IoT security's delicate intricacies, including obtaining discernment of the distinctive susceptibilities of different IoT devices and the imperative of preserving such devices with consistent refreshes and protection fixes to mitigate any potential vulnerabilities.

Overall, it is crucial to secure all layers of healthcare IoT systems to safeguard patient data, prevent cyberattacks, and ensure patient safety. Healthcare organizations must take proactive steps to secure their IoT systems, stay informed of the latest security threats and best practices, and implement appropriate security controls at all layers of the system. By doing so, the IoT perception layer can continue to enable the development of innovative healthcare technologies that improve patient outcomes and enhance the quality of care. The National Institute of Standards and Technology guidelines for securing IoT devices in healthcare settings can provide a useful framework for healthcare organizations to improve their security posture. Collaboration between different stakeholders is essential to address the challenges of securing healthcare IoT systems and protecting patient data and safety.

The importance of security and privacy in healthcare IoT cannot be overstated. The systematic literature review highlights that healthcare IoT involves sensitive and personal data, such as patient's health records and biometric data, and breaches can have severe consequences for patients, healthcare providers, and organizations. For example, data breaches can result in identity theft, financial loss, reputational damage, and legal liabilities. Furthermore, compromised healthcare IoT devices can lead to medical errors, delayed diagnoses, and incorrect treatments. Therefore, ensuring the security and privacy of healthcare IoT should be a top priority for all stakeholders involved.

The review emphasizes the need for comprehensive security measures that address security and privacy issues and attacks at multiple layers of the IoT architecture. Single-layer security measures, such as firewalls or encryption, may not be sufficient to protect against sophisticated attacks that exploit vulnerabilities at different layers. Therefore, organizations must adopt a multilayered approach that includes physical security, network security, data security, and access control. This approach should also include periodic

risk assessments, vulnerability scans, and penetration testing to identify and address security gaps.

The review highlights that stakeholders have a shared responsibility to ensure the security and privacy of healthcare IoT. Healthcare organizations, device manufacturers, and software developers must work collaboratively to identify and address security and privacy issues and attacks. This requires implementing security-by-design principles, such as threat modeling, secure coding practices, and regular software updates. Security-by-design principles enable software developers to critically analyze and identify the security needs of the software systems [156]. Subsequently, these security needs can be embedded into the software architecture, so that the software developers can make sure that the developed software systems include all necessary security controls. Healthcare organizations should also provide training and education to staff and end-users to ensure that security and privacy protocols are followed. Healthcare organizations should follow a zero-trust approach, where they understand that the networks are vulnerable to security threats all the time and there are many internal and external security threats in a network regularly. Thus, healthcare organizations should follow the basic principles of the zero-trust approach such as taking necessary measures to ensure user authenticity and allowing users to access company resources only using trusted devices [157]. Additionally, regulatory bodies and policy makers must collaborate with industry stakeholders to establish and enforce standards and regulations that promote the security and privacy of healthcare IoT. These standards should encompass rigorous data privacy regulations to safeguard sensitive patient information [158], robust security certification programs to ensure the integrity of IoT devices and networks [159], and clear incident reporting requirements to address any security breaches or vulnerabilities promptly [160]. Furthermore, it is essential to strike a delicate balance between regulation enforcement and technological advancement to foster innovation in the healthcare IoT sector. Overregulation could stifle innovation and impede the development of new technologies that could significantly benefit patient care and healthcare delivery. Therefore, regulatory frameworks should be flexible enough to accommodate technological advancements while still prioritizing patient safety and privacy [161]. Collaboration among industry stakeholders, regulatory bodies, policy makers, and cybersecurity experts is crucial to developing effective and adaptable regulatory measures that mitigate risks without hindering progress [162].

Our review suggests that ensuring the security and privacy of healthcare IoT requires a coordinated effort among all stakeholders. This includes adopting a multilayered security approach, implementing security-by-design principles, following the basic principles of the zero-trust approach, providing training and education, and complying with relevant regulations and standards. By doing so, stakeholders can mitigate the risk of security and privacy issues and attacks and enhance the trust and reliability of healthcare IoT.

The review identifies several gaps in the current research on IoT security and privacy issues and attacks in healthcare. First, most studies have focused on individual layers of the IoT architecture, rather than examining the entire IoT system. Second, there is a lack of empirical research that evaluates the effectiveness of different security measures and strategies in mitigating security and privacy risks. Third, there is a need for more research on the human factor in IoT security, including end-user behavior, security awareness, and training. Fourth, there is a lack of research on the ethical implications of IoT security and privacy, particularly concerning data ownership, consent, and transparency [163].

To address the gaps in the current literature, future research on IoT security and privacy issues and attacks in healthcare should adopt a more holistic and multidisciplinary approach. This includes examining the entire IoT system, including the devices, network, and applications, and considering the socio-technical and ethical aspects of IoT security and privacy [163]. Future research should also employ more rigorous and empirical research methods, such as experiments and field studies, to evaluate the effectiveness of different security measures and strategies. Additionally, research should explore the role of end-users in IoT security and privacy and identify effective training and awareness

programs. Finally, research should investigate the ethical implications of IoT security and privacy and develop frameworks and guidelines for ensuring data ownership, consent, and transparency.

In addition to the above, there are several areas for further exploration of IoT security and privacy issues and attacks in healthcare. One area is the use of artificial intelligence and machine learning in enhancing IoT security and privacy, as suggested by [164]. Future research exploration should explore the development of inherently flexible security technologies that are amenable to the continuous evolution of IoT systems and how their applications in healthcare are developing. This includes investigating AI-driven security methods that can predict and mitigate potential threats in real-time.

Another area is the impact of emerging technologies, such as 5G and edge computing, on IoT security and privacy. Additionally, research should examine the security and privacy challenges and opportunities of interoperability and data sharing in healthcare IoT. Finally, research should explore the international and cross-cultural aspects of IoT security and privacy, including the role of cultural values and norms in shaping security and privacy attitudes and behaviors.

6. Conclusions and Future Work

In this review, the security and privacy issues and attacks in healthcare IoT were evaluated according to the perception layer, the network layer, and the application layer. This study revealed various threats, such as unauthorized access, data breaches, and DOS attacks, with potentially serious consequences for healthcare providers, organizations, and patients. The review highlighted various security measures, such as encryption, firewalls, and access control, to address these risks.

This study has practical implications, emphasizing that healthcare organizations, software developers, and device manufacturers must prioritize healthcare IoT security and privacy. A comprehensive, multilayered security approach, security-by-design principles, zero-trust principles, and training for staff and end-users must be adopted. Regulators and policy makers must also establish and enforce standards and regulations that promote the security and privacy of healthcare IoT.

This study identified gaps in current literature and areas for further exploration, such as the need for more holistic and empirical research that considers the socio-technical and ethical aspects of IoT security and privacy, the role of emerging technologies, and international and cross-cultural factors that affect security and privacy attitudes and behaviors. Overall, this study underscores the importance of ensuring the security and privacy of healthcare IoT, with stakeholders' coordinated efforts to address the complex and evolving security and privacy threats in this field. This can enhance healthcare IoT trust and reliability, reduce the risks of security and privacy issues and attacks, and ultimately improve healthcare delivery quality and safety.

Author Contributions: The idea of this research is conceptualized by S.M. S.M. developed the research method protocol and involved data collection with T.C. T.C. carried out the data analysis and prepared the first draft of this report under S.M.'s supervision and guidance. The report and findings were enhanced by M.S. contribution. D.A., S.Y. and F.H. provided some technical insights and contributed to revising, editing and formatting this manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by AUT, School of Engineering, Computer and Mathematical Sciences, 2022 Summer Research Scholarships.

Data Availability Statement: All the data to conduct this research were collected from the scientific databases as discussed in the Section 3 and available online.

Acknowledgments: The authors acknowledge the use of AI for English editing. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the contents of the publication.

Conflicts of Interest: Author David Airehrour is employed by the company Together Communications, Auckland 1010, New Zealand. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Sisodia, R.C.; Dankers, C.; Orav, J.; Joseph, B.; Meyers, P.; Wright, P.; Amand, D.S.; Del Carmen, M.; Ferris, T.; Heng, M. Factors Associated With Increased Collection of Patient-Reported Outcomes Within a Large Health Care System. *JAMA Netw. Open* **2020**, *3*, e202764. [CrossRef] [PubMed]
2. Luo, E.; Bhuiyan, M.Z.A.; Wang, G.; Rahman, M.A.; Wu, J.; Atiquzzaman, M. Privacyprotector: Privacy-protected patient data collection in IoT-based healthcare systems. *IEEE Commun. Mag.* **2018**, *56*, 163–168. [CrossRef]
3. Madanian, S.; Parry, D. Identifying the Potential of RFID in Disaster Healthcare: An International Delphi Study. *Electronics* **2021**, *10*, 2621. [CrossRef]
4. Kiran, D.R. Chapter 35—Internet of Things. In *Production Planning and Control*; Kiran, D.R., Ed.; Butterworth-Heinemann: Oxford, UK, 2019; pp. 495–513.
5. Zhao, K.; Ge, L. A Survey on the Internet of Things Security. In Proceedings of the 2013 Ninth International Conference on Computational Intelligence and Security, Emeishan, China, 14–15 December 2013; pp. 663–667.
6. Al-khafajiy, M.; Baker, T.; Chalmers, C.; Asim, M.; Kolivand, H.; Fahim, M.; Waraich, A. Remote health monitoring of elderly through wearable sensors. *Multimed. Tools Appl.* **2019**, *78*, 24681–24706. [CrossRef]
7. Albahri, A.S.; Alwan, J.K.; Taha, Z.K.; Ismail, S.F.; Hamid, R.A.; Zaidan, A.A.; Albahri, O.S.; Zaidan, B.B.; Alamoodi, A.H.; Alsalem, M.A. IoT-based telemedicine for disease prevention and health promotion: State-of-the-Art. *J. Netw. Comput. Appl.* **2021**, *173*, 102873. [CrossRef]
8. Riad, K.; Hamza, R.; Yan, H. Sensitive and Energetic IoT Access Control for Managing Cloud Electronic Health Records. *IEEE Access* **2019**, *7*, 86384–86393. [CrossRef]
9. Vistro, D.M.; Rehman, A.u.; Mehmood, S.; Idrees, M.S.; Munawar, A. An IoT-Based Approach for Smart Ambulance Service Using Thingspeak Cloud. *J. Crit. Rev.* **2020**, *7*, 1697–1703.
10. Yin, X.C.; Liu, Z.G.; Ndibanje, B.; Nkenyereye, L.; Riazul Islam, S. An IoT-based anonymous function for security and privacy in healthcare sensor networks. *Sensors* **2019**, *19*, 3146. [CrossRef]
11. Usak, M.; Kubiatko, M.; Shabbir, M.S.; Viktorovna Dudnik, O.; Jermsittiparsert, K.; Rajabion, L. Health care service delivery based on the Internet of things: A systematic and comprehensive study. *Int. J. Commun. Syst.* **2020**, *33*, e4179. [CrossRef]
12. Bhatt, V.; Chakraborty, S. Improving service engagement in healthcare through internet of things based healthcare systems. *J. Sci. Technol. Policy Manag.* **2023**, *14*, 53–73. [CrossRef]
13. Khattak, H.A.; Shah, M.A.; Khan, S.; Ali, I.; Imran, M. Perception layer security in Internet of Things. *Future Gener. Comput. Syst.* **2019**, *100*, 144–164. [CrossRef]
14. Rghioui, A.; Oumnad, A. Internet of things: Surveys for measuring human activities from everywhere. *Int. J. Electr. Comput. Eng.* **2017**, *7*, 2474.
15. Mouha, R.A. Internet of things (IoT). *J. Data Anal. Inf. Process.* **2021**, *9*, 77–101.
16. Shouran, Z.; Ashari, A.; Priyambodo, T. Internet of things (IoT) of smart home: Privacy and security. *Int. J. Comput. Appl.* **2019**, *182*, 3–8.
17. Chanal, P.M.; Kakkasageri, M.S. Security and Privacy in IoT: A Survey. *Wirel. Pers. Commun.* **2020**, *115*, 1667–1693. [CrossRef]
18. Kumar, S.; Tiwari, P.; Zymbler, M. Internet of Things is a revolutionary approach for future technology enhancement: A review. *J. Big Data* **2019**, *6*, 111. [CrossRef]
19. Al-Sarawi, S.; Anbar, M.; Alieyan, K.; Alzubaidi, M. Internet of Things (IoT) communication protocols: Review. In Proceedings of the 2017 8th International Conference on Information Technology (ICIT), Amman, Jordan, 17–18 May 2017; pp. 685–690.
20. Sanctis, M.D.; Cianca, E.; Araniti, G.; Bisio, I.; Prasad, R. Satellite Communications Supporting Internet of Remote Things. *IEEE Internet Things J.* **2016**, *3*, 113–123. [CrossRef]
21. Ma, Z.; Xiao, M.; Xiao, Y.; Pang, Z.; Poor, H.V.; Vucetic, B. High-Reliability and Low-Latency Wireless Communication for Internet of Things: Challenges, Fundamentals, and Enabling Technologies. *IEEE Internet Things J.* **2019**, *6*, 7946–7970. [CrossRef]
22. Yassein, M.B.; Shatnawi, M.Q.; Al-zoubi, D. Application layer protocols for the Internet of Things: A survey. In Proceedings of the 2016 International Conference on Engineering & MIS (ICEMIS), Agadir, Morocco, 22–24 September 2016; pp. 1–4.
23. Hussain, A.; Ali, T.; Althobiani, F.; Draz, U.; Irfan, M.; Yasin, S.; Shafiq, S.; Safdar, Z.; Glowacz, A.; Nowakowski, G.; et al. Security Framework for IoT Based Real-Time Health Applications. *Electronics* **2021**, *10*, 719. [CrossRef]
24. Kavre, M.; Gadekar, A.; Gadhade, Y. Internet of Things (IoT): A Survey. In Proceedings of the 2019 IEEE Pune Section International Conference (PuneCon), Pune, India, 18–20 December 2019; pp. 1–6.
25. Ahmed, Z.; Mohamed, K.; Zeeshan, S.; Dong, X. Artificial intelligence with multi-functional machine learning platform development for better healthcare and precision medicine. *Database* **2020**, *2020*, baaa010. [CrossRef]
26. Khatoon, N.; Roy, S.; Pranav, P. A Survey on Applications of Internet of Things in Healthcare. In *Internet of Things and Big Data Applications: Recent Advances and Challenges*; Balas, V.E., Solanki, V.K., Kumar, R., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 89–106.

27. Abiodun, O.I.; Abiodun, E.O.; Alawida, M.; Alkhawaldeh, R.S.; Arshad, H. A Review on the Security of the Internet of Things: Challenges and Solutions. *Wirel. Pers. Commun.* **2021**, *119*, 2603–2637. [CrossRef]
28. Moore, W.; Frye, S. Review of HIPAA, part 1: History, protected health information, and privacy and security rules. *J. Nucl. Med. Technol.* **2019**, *47*, 269–272. [CrossRef] [PubMed]
29. Hathaliya, J.J.; Tanwar, S. An exhaustive survey on security and privacy issues in Healthcare 4.0. *Comput. Commun.* **2020**, *153*, 311–335. [CrossRef]
30. De Michele, R.; Furini, M. Iot healthcare: Benefits, issues and challenges. In Proceedings of the 5th EAI International Conference on Smart Objects and Technologies for Social Good, Valencia, Spain, 25–27 September 2019; pp. 160–164.
31. Appari, A.; Johnson, M.E. Information security and privacy in healthcare: Current state of research. *Int. J. Internet Enterp. Manag.* **2010**, *6*, 279–314. [CrossRef]
32. Agaku, I.T.; Ayo-Yusuf, O.A.; Connolly, G.N.; Adisa, A.O. Concern about security and privacy, and perceived control over collection and use of health information are related to withholding of health information from healthcare providers. *J. Am. Med. Inform. Assoc.* **2014**, *21*, 374–378. [CrossRef]
33. Zhou, W.; Jia, Y.; Peng, A.; Zhang, Y.; Liu, P. The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, and Challenges Yet to Be Solved. *IEEE Internet Things J.* **2019**, *6*, 1606–1616. [CrossRef]
34. Maswadi, K.; Ghani, N.B.A.; Hamid, S.B. Systematic Literature Review of Smart Home Monitoring Technologies Based on IoT for the Elderly. *IEEE Access* **2020**, *8*, 92244–92261. [CrossRef]
35. Azeez, N.A.; der Vyver, C.V. Security and privacy issues in e-health cloud-based system: A comprehensive content analysis. *Egypt. Inform. J.* **2019**, *20*, 97–108. [CrossRef]
36. Chacko, A.; Hayajneh, T. Security and privacy issues with IoT in healthcare. *EAI Endorsed Trans. Pervasive Health Technol.* **2018**, *4*, e2. [CrossRef]
37. HIPAA. Healthcare Data Breach Statistics. Available online: https://www.hipaajournal.com/healthcare-data-breach-statistics/ (accessed on 20 October 2023).
38. Centers for Disease Control and Prevention. Health Insurance Portability and Accountability Act of 1996 (HIPAA). Available online: https://www.cdc.gov/phlp/php/resources/health-insurance-portability-and-accountability-act-of-1996-hipaa.html (accessed on 20 October 2023).
39. Pramanik, P.K.D.; Pareek, G.; Nayyar, A. Chapter 14—Security and Privacy in Remote Healthcare: Issues, Solutions, and Standards. In *Telemedicine Technologies*; Jude, H.D., Balas, V.E., Eds.; Academic Press: Cambridge, MA, USA, 2019; pp. 201–225.
40. Awotunde, J.B.; Jimoh, R.G.; Folorunso, S.O.; Adeniyi, E.A.; Abiodun, K.M.; Banjo, O.O. Privacy and Security Concerns in IoT-Based Healthcare Systems. In *The Fusion of Internet of Things, Artificial Intelligence, and Cloud Computing in Health Care*; Siarry, P., Jabbar, M.A., Aluvalu, R., Abraham, A., Madureira, A., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 105–134.
41. Butpheng, C.; Yeh, K.-H.; Xiong, H. Security and Privacy in IoT-Cloud-Based e-Health Systems—A Comprehensive Review. *Symmetry* **2020**, *12*, 1191. [CrossRef]
42. Coventry, L.; Branley, D. Cybersecurity in healthcare: A narrative review of trends, threats and ways forward. *Maturitas* **2018**, *113*, 48–52. [CrossRef] [PubMed]
43. Sadek, I.; Rehman, S.U.; Codjo, J.; Abdulrazak, B. Privacy and security of IoT based healthcare systems: Concerns, solutions, and recommendations. In Proceedings of the How AI Impacts Urban Living and Public Health: 17th International Conference, ICOST 2019, New York City, NY, USA, 14–16 October 2019; Proceedings 17, 2019. pp. 3–17.
44. Paulsen, C. *Glossary of Key Information Security Terms*; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2018.
45. Luh, F.; Yen, Y. Cybersecurity in Science and Medicine: Threats and Challenges. *Trends Biotechnol.* **2020**, *38*, 825–828. [CrossRef] [PubMed]
46. Semantha, F.H.; Azam, S.; Yeo, K.C.; Shanmugam, B. A Systematic Literature Review on Privacy by Design in the Healthcare Sector. *Electronics* **2020**, *9*, 452. [CrossRef]
47. Keshta, I.; Odeh, A. Security and privacy of electronic health records: Concerns and challenges. *Egypt. Inform. J.* **2021**, *22*, 177–183. [CrossRef]
48. Yeo, L.H.; Banfield, J. Human Factors in Electronic Health Records Cybersecurity Breach: An Exploratory Analysis. *Perspect. Health Inf. Manag.* **2022**, *19*, 1i. [PubMed]
49. Seh, A.H.; Zarour, M.; Alenezi, M.; Sarkar, A.K.; Agrawal, A.; Kumar, R.; Ahmad Khan, R. Healthcare Data Breaches: Insights and Implications. *Healthcare* **2020**, *8*, 133. [CrossRef]
50. McGraw, D.; Mandl, K.D. Privacy protections to encourage use of health-relevant digital data in a learning health system. *NPJ Digit. Med.* **2021**, *4*, 2. [CrossRef]
51. Abouelmehdi, K.; Beni-Hessane, A.; Khaloufi, H. Big healthcare data: Preserving security and privacy. *J. Big Data* **2018**, *5*, 1. [CrossRef]
52. Robinson, J. US Healthcare Data Breach Statistics. Available online: https://www.privacyaffairs.com/healthcare-data-breach-statistics/ (accessed on 15 October 2023).

53. Argaw, S.T.; Troncoso-Pastoriza, J.R.; Lacey, D.; Florin, M.-V.; Calcavecchia, F.; Anderson, D.; Burleson, W.; Vogel, J.-M.; O'Leary, C.; Eshaya-Chauvin, B.; et al. Cybersecurity of Hospitals: Discussing the challenges and working towards mitigating the risks. *BMC Med. Inform. Decis. Mak.* **2020**, *20*, 146. [CrossRef]
54. Elhoseny, M.; Thilakarathne, N.N.; Alghamdi, M.I.; Mahendran, R.K.; Gardezi, A.A.; Weerasinghe, H.; Welhenge, A. Security and Privacy Issues in Medical Internet of Things: Overview, Countermeasures, Challenges and Future Directions. *Sustainability* **2021**, *13*, 11645. [CrossRef]
55. Straits Research. IoT Healthcare Market Size Is Projected to Reach USD 486.34 Billion by 2031, Growing at a CAGR of 19.27%: Straits Research. Available online: https://www.globenewswire.com/en/news-release/2022/11/15/2556464/0/en/IoT-Healthcare-Market-Size-is-projected-to-reach-USD-486-34-Billion-by-2031-growing-at-a-CAGR-of-19-27-Straits-Research.html (accessed on 15 October 2023).
56. Kadhim, K.T.; Alsahlany, A.M.; Wadi, S.M.; Kadhum, H.T. An Overview of Patient's Health Status Monitoring System Based on Internet of Things (IoT). *Wirel. Pers. Commun.* **2020**, *114*, 2235–2262. [CrossRef]
57. Weenk, M.; Bredie, S.J.; Koeneman, M.; Hesselink, G.; van Goor, H.; van de Belt, T.H. Continuous Monitoring of Vital Signs in the General Ward Using Wearable Devices: Randomized Controlled Trial. *J. Med. Internet Res.* **2020**, *22*, e15471. [CrossRef] [PubMed]
58. Al-Rawashdeh, M.; Keikhosrokiani, P.; Belaton, B.; Alawida, M.; Zwiri, A. IoT Adoption and Application for Smart Healthcare: A Systematic Review. *Sensors* **2022**, *22*, 5377. [CrossRef] [PubMed]
59. Babar, S.; Mahalle, P.; Stango, A.; Prasad, N.; Prasad, R. Proposed Security Model and Threat Taxonomy for the Internet of Things (IoT). In *Recent Trends in Network Security and Applications: Third International Conference, CNSA 2010, Chennai, India, 23–25 July 2010. Proceedings*; Meghanathan, N., Boumerdassi, S., Chaki, N., Nagamalai, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; pp. 420–429.
60. Al Ameen, M.; Liu, J.; Kwak, K. Security and Privacy Issues in Wireless Sensor Networks for Healthcare Applications. *J. Med. Syst.* **2012**, *36*, 93–101. [CrossRef]
61. Jing, Q.; Lu, J.; Qiu, D.; Vasilakos, A.V.; Wan, J. Security of the Internet of Things: Perspectives and challenges. *Wirel. Netw.* **2014**, *20*, 2481–2501. [CrossRef]
62. Watzlaf, V.J.M.; Zhou, L.; DeAlmeida, D.R.; Hartman, L.M. A Systematic Review of Research Studies Examining Telehealth Privacy and Security Practices Used by Healthcare Providers. *Int. J. Telerehabilitation* **2017**, *9*, 39–58. [CrossRef]
63. Abouzakhar, N.S.; Jones, A.; Angelopoulou, O. Internet of Things Security: A Review of Risks and Threats to Healthcare Sector. In Proceedings of the 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Exeter, UK, 21–23 June 2017; pp. 373–378.
64. Newaz, A.I.; Sikder, A.K.; Rahman, M.A.; Uluagac, A.S. A Survey on Security and Privacy Issues in Modern Healthcare Systems: Attacks and Defenses. *ACM Trans. Comput. Healthc.* **2021**, *2*, 27. [CrossRef]
65. Papaioannou, M.; Karageorgou, M.; Mantas, G.; Sucasas, V.; Essop, I.; Rodriguez, J.; Lymberopoulos, D. A Survey on Security Threats and Countermeasures in Internet of Medical Things (IoMT). *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e4049. [CrossRef]
66. Sun, W.; Cai, Z.; Li, Y.; Liu, F.; Fang, S.; Wang, G. Security and Privacy in the Medical Internet of Things: A Review. *Secur. Commun. Netw.* **2018**, *2018*, 5978636. [CrossRef]
67. Snyder, H. Literature review as a research methodology: An overview and guidelines. *J. Bus. Res.* **2019**, *104*, 333–339. [CrossRef]
68. Selvaraj, S.; Sundaravaradhan, S. Challenges and opportunities in IoT healthcare systems: A systematic review. *SN Appl. Sci.* **2019**, *2*, 139. [CrossRef]
69. Abomhara, M.; Køien, G.M. Security and privacy in the Internet of Things: Current status and open issues. In Proceedings of the 2014 International Conference on Privacy and Security in Mobile Systems (PRISMS), Aalborg, Denmark, 11–14 May 2014; pp. 1–8.
70. Vignesh, M.R.; Sivakumar, S. Healthcare sensors issues, challenges & security threats in wireless body area network: A comprehensive survey. *Int. J. Trend Sci. Res. Dev* **2021**, *5*, 989–997.
71. Shahid, J.; Ahmad, R.; Kiani, A.K.; Ahmad, T.; Saeed, S.; Almuhaideb, A.M. Data Protection and Privacy of the Internet of Healthcare Things (IoHTs). *Appl. Sci.* **2022**, *12*, 1927. [CrossRef]
72. BajriĆ, S. Data Security and Privacy Issues in Healthcare. *Appl. Med. Inform.* **2020**, *42*, 19–27.
73. Alkhatib, S.; Waycott, J.; Buchanan, G.; Bosua, R. Privacy and the Internet of Things (IoT) Monitoring Solutions for Older Adults: A Review. *Stud. Health Technol. Inform.* **2018**, *252*, 8–14.
74. Djenna, A.; Saïdouni, D.E. Cyber Attacks Classification in IoT-Based-Healthcare Infrastructure. In Proceedings of the 2018 2nd Cyber Security in Networking Conference (CSNet), Paris, France, 24–26 October 2018; pp. 1–4.
75. Li, C.; Wang, J.; Wang, S.; Zhang, Y. A review of IoT applications in healthcare. *Neurocomputing* **2024**, *565*, 127017. [CrossRef]
76. Zakaria, H.; Abu Bakar, N.A.; Hassan, N.H.; Yaacob, S. IoT Security Risk Management Model for Secured Practice in Healthcare Environment. *Procedia Comput. Sci.* **2019**, *161*, 1241–1248. [CrossRef]
77. Hossain, M.M.; Fotouhi, M.; Hasan, R. Towards an Analysis of Security Issues, Challenges, and Open Problems in the Internet of Things. In Proceedings of the 2015 IEEE World Congress on Services, New York, NY, USA, 27 June–2 July 2015; pp. 21–28.
78. Abie, H.; Balasingham, I. Risk-based adaptive security for smart IoT in eHealth. In Proceedings of the Proceedings of the 7th International Conference on Body Area Networks, Oslo, Norway, 24–26 February 2012; pp. 269–275.
79. Li, X.; Dai, H.-N.; Wang, Q.; Imran, M.; Li, D.; Imran, M.A. Securing Internet of Medical Things with Friendly-jamming schemes. *Comput. Commun.* **2020**, *160*, 431–442. [CrossRef]

80. Yaacoub, J.-P.A.; Noura, M.; Noura, H.N.; Salman, O.; Yaacoub, E.; Couturier, R.; Chehab, A. Securing internet of medical things systems: Limitations, issues and recommendations. *Future Gener. Comput. Syst.* **2020**, *105*, 581–606. [CrossRef]
81. Kumar, P.; Lee, H.J. Security issues in healthcare applications using wireless medical sensor networks: A survey. *Sensors* **2012**, *12*, 55–91. [CrossRef]
82. Suo, H.; Wan, J.; Zou, C.; Liu, J. Security in the Internet of Things: A Review. In Proceedings of the 2012 International Conference on Computer Science and Electronics Engineering, Hangzhou, China, 23–25 March 2012; pp. 648–651.
83. Priestman, W.; Anstis, T.; Sebire, I.G.; Sridharan, S.; Sebire, N.J. Phishing in healthcare organisations: Threats, mitigation and approaches. *BMJ Health Care Inform.* **2019**, *26*, e100031. [CrossRef] [PubMed]
84. Waheed, N.; He, X.; Ikram, M.; Usman, M.; Hashmi, S.S.; Usman, M. Security and Privacy in IoT Using Machine Learning and Blockchain: Threats and Countermeasures. *ACM Comput. Surv.* **2020**, *53*, 122. [CrossRef]
85. Chen, F.; Huang, J.; Wang, C.; Tang, Y.; Huang, C.; Xie, D.; Wang, T.; Zhao, C. Data Access Control Based on Blockchain in Medical Cyber Physical Systems. *Secur. Commun. Netw.* **2021**, *2021*, 3395537. [CrossRef]
86. Hasan, M.K.; Ghazal, T.M.; Saeed, R.A.; Pandey, B.; Gohel, H.; Eshmawi, A.A.; Abdel-Khalek, S.; Alkhassawneh, H.M. A review on security threats, vulnerabilities, and counter measures of 5G enabled Internet-of-Medical-Things. *IET Commun.* **2022**, *16*, 421–432. [CrossRef]
87. Alromaihi, S.; Elmedany, W.; Balakrishna, C. Cyber Security Challenges of Deploying IoT in Smart Cities for Healthcare Applications. In Proceedings of the 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), Barcelona, Spain, 6–8 August 2018; pp. 140–145.
88. Wasserman, L.; Wasserman, Y. Hospital cybersecurity risks and gaps: Review (for the non-cyber professional). *Front. Digit. Health* **2022**, *4*, 862221. [CrossRef]
89. Chattopadhyay, A.K.; Nag, A.; Ghosh, D.; Chanda, K. A Secure Framework for IoT-Based Healthcare System. In Proceedings of the International Ethical Hacking Conference 2018; Springer: Singapore, 2019; pp. 383–393.
90. Mamdouh, M.; Awad, A.I.; Khalaf, A.A.M.; Hamed, H.F.A. Authentication and Identity Management of IoHT Devices: Achievements, Challenges, and Future Directions. *Comput. Secur.* **2021**, *111*, 102491. [CrossRef]
91. Abdullah, A.; Hamad, R.; Abdulrahman, M.; Moala, H.; Elkhediri, S. CyberSecurity: A Review of Internet of Things (IoT) Security Issues, Challenges and Techniques. In Proceedings of the 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 1–3 May 2019; pp. 1–6.
92. Alsubaei, F.; Abuhussein, A.; Shiva, S. Security and Privacy in the Internet of Medical Things: Taxonomy and Risk Assessment. In Proceedings of the 2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops), Singapore, 9 October 2017; pp. 112–120.
93. Gupta, B.B.; Dahiya, A. *Distributed Denial of Service (DDoS) Attacks: Classification, Attacks, Challenges and Countermeasures*; CRC Press: Boca Raton, FL, USA, 2021.
94. Bediya, A.K.; Kumar, R. A Novel Intrusion Detection System for Internet of Things Network Security. In *Research Anthology on Convergence of Blockchain, Internet of Things, and Security*; Management Association, I.R., Ed.; IGI Global: Hershey, PA, USA, 2023; pp. 330–348.
95. Sharma, K.; Gupta, B.B. Taxonomy of Distributed Denial of Service (DDoS) Attacks and Defense Mechanisms in Present Era of Smartphone Devices. *Int. J. E-Serv. Mob. Appl. (IJESMA)* **2018**, *10*, 58–74. [CrossRef]
96. Ray, S.; Mishra, K.N.; Dutta, S. Detection and prevention of DDoS attacks on M-healthcare sensitive data: A novel approach. *Int. J. Inf. Technol.* **2022**, *14*, 1333–1341. [CrossRef]
97. Alagar, V.; Alsaig, A.; Ormandjiva, O.; Wan, K. Context-Based Security and Privacy for Healthcare IoT. In Proceedings of the 2018 IEEE International Conference on Smart Internet of Things (SmartIoT), Xi'an, China, 17–19 August 2018; pp. 122–128.
98. Mahjabin, T.; Xiao, Y.; Sun, G.; Jiang, W. A survey of distributed denial-of-service attack, prevention, and mitigation techniques. *Int. J. Distrib. Sens. Netw.* **2017**, *13*, 1550147717741463. [CrossRef]
99. Imdad, M.; Jacob, D.W.; Mahdin, H.; Baharum, Z.; Shaharudin, S.M.; Azmi, M.S. Internet of things (IoT); security requirements, attacks and counter measures. *Indones. J. Electr. Eng. Comput. Sci.* **2020**, *18*, 1520–1530.
100. Shah, S.; Simnani, S.S.A.; Banday, M.T. A Study of Security Attacks on Internet of Things and Its Possible Solutions. In Proceedings of the 2018 International Conference on Automation and Computational Engineering (ICACE), Noida, India, 3–4 October 2018; pp. 203–209.
101. Obaidat, M.A.; Obeidat, S.; Holst, J.; Al Hayajneh, A.; Brown, J. A Comprehensive and Systematic Survey on the Internet of Things: Security and Privacy Challenges, Security Frameworks, Enabling Technologies, Threats, Vulnerabilities and Countermeasures. *Computers* **2020**, *9*, 44. [CrossRef]
102. Nifakos, S.; Chandramouli, K.; Nikolaou, C.K.; Papachristou, P.; Koch, S.; Panaousis, E.; Bonacina, S. Influence of Human Factors on Cyber Security within Healthcare Organisations: A Systematic Review. *Sensors* **2021**, *21*, 5119. [CrossRef]
103. Alwan, Z.S.; Younis, M.F. Detection and prevention of SQL injection attack: A survey. *Int. J. Comput. Sci. Mob. Comput.* **2017**, *6*, 5–17.
104. Rajendran, G.; Nivash, R.S.R.; Parthy, P.P.; Balamurugan, S. Modern security threats in the Internet of Things (IoT): Attacks and Countermeasures. In Proceedings of the 2019 International Carnahan Conference on Security Technology (ICCST), Chennai, India, 1–3 October 2019; pp. 1–6.

105. Tukur, Y.M.; Ali, Y.S. Demonstrating the Effect of Insider Attacks on Perception Layer of Internet of Things (IoT) Systems. In Proceedings of the 2019 15th International Conference on Electronics, Computer and Computation (ICECCO), Abuja, Nigeria, 10–12 December 2019; pp. 1–6.
106. Lee, I. Analysis of Insider Threats in the Healthcare Industry: A Text Mining Approach. *Information* **2022**, *13*, 404. [CrossRef]
107. Alsowail, R.A.; Al-Shehari, T. Techniques and countermeasures for preventing insider threats. *PeerJ Comput. Sci.* **2022**, *8*, e938. [CrossRef]
108. Albalawi, A.M.; Almaiah, M.A. Assessing and reviewing of cyber-security threats, attacks, mitigation techniques in IoT environment. *J. Theor. Appl. Inf. Technol* **2022**, *100*, 2988–3011.
109. Farhin, F.; Kaiser, M.S.; Mahmud, M. Towards Secured Service Provisioning for the Internet of Healthcare Things. In Proceedings of the 2020 IEEE 14th International Conference on Application of Information and Communication Technologies (AICT), Tashkent, Uzbekistan, 7–9 October 2020; pp. 1–6.
110. Rughoobur, P.; Nagowah, L.; Rughoobur, P.; Nagowah, L. A lightweight replay attack detection framework for battery depended IoT devices designed for healthcare. In Proceedings of the 2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions) (ICTUS), Dubai, United Arab Emirates, 18–20 December 2017; pp. 811–817.
111. What Is RYUK Ransomware? Available online: https://www.trendmicro.com/en_nz/what-is/ransomware/ryuk-ransomware.html#:~:text=Ryuk%20is%20ransomware%20version%20attributed,by%20the%20end%20of%202020 (accessed on 10 December 2023).
112. Swessi, D.; Idoudi, H. A Survey on Internet-of-Things Security: Threats and Emerging Countermeasures. *Wirel. Pers. Commun.* **2022**, *124*, 1557–1592. [CrossRef]
113. Liu, X.; Qian, C.; Hatcher, W.G.; Xu, H.; Liao, W.; Yu, W. Secure Internet of Things (IoT)-Based Smart-World Critical Infrastructures: Survey, Case Study and Research Opportunities. *IEEE Access* **2019**, *7*, 79523–79544. [CrossRef]
114. Mahanty, A.; Singh, G.; Som, S.; Khatri, S.K. Security Issues and Challenges in Perception Layer of Smart Healthcare. In Proceedings of the 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 29–31 August 2018; pp. 639–644.
115. Ren, Z.; Liu, X.; Ye, R.; Zhang, T. Security and privacy on internet of things. In Proceedings of the 2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC), Macau, China, 21–23 July 2017; pp. 140–144.
116. Shrivastava, V.; Namdev, M. A Review on Security and Privacy Issues in Wireless Body Area Networks for Healthcare Applications. *Smart Moves J. Ijoscience* **2019**, *5*, 22–28. [CrossRef]
117. Mrabet, H.; Belguith, S.; Alhomoud, A.; Jemai, A. A Survey of IoT Security Based on a Layered Architecture of Sensing and Data Analysis. *Sensors* **2020**, *20*, 3625. [CrossRef]
118. Habibzadeh, H.; Dinesh, K.; Shishvan, O.R.; Boggio-Dandry, A.; Sharma, G.; Soyata, T. A Survey of Healthcare Internet of Things (HIoT): A Clinical Perspective. *IEEE Internet Things J.* **2020**, *7*, 53–71. [CrossRef] [PubMed]
119. Bakar, N.A.A.; Ramli, W.M.W.; Hassan, N.H. The internet of things in healthcare: An overview, challenges and model plan for security risks management process. *Indones. J. Electr. Eng. Comput. Sci. (IJEECS)* **2019**, *15*, 414–420.
120. Amaraweera, S.P.; Halgamuge, M.N. Internet of Things in the Healthcare Sector: Overview of Security and Privacy Issues. In *Security, Privacy and Trust in the IoT Environment*; Mahmood, Z., Ed.; Springer International Publishing: Cham, Switzerland, 2019; pp. 153–179.
121. Fazeldehkordi, E.; Owe, O.; Noll, J. Security and Privacy in IoT Systems: A Case Study of Healthcare Products. In Proceedings of the 2019 13th International Symposium on Medical Information and Communication Technology (ISMICT), Oslo, Norway, 8–10 May 2019; pp. 1–8.
122. Perwej, Y.; Akhtar, N.; Kulshrestha, N.; Mishra, P. A Methodical Analysis of Medical Internet of Things (MIoT) Security and Privacy in Current and Future Trends. *J. Emerg. Technol. Innov. Res.* **2022**, *9*, d346–d371.
123. Islam, M.M.; Nooruddin, S.; Karray, F.; Muhammad, G. Internet of Things: Device Capabilities, Architectures, Protocols, and Smart Applications in Healthcare Domain. *IEEE Internet Things J.* **2023**, *10*, 3611–3641. [CrossRef]
124. Pandey, P.; Pandey, S.C.; Kumar, U. Security issues of internet of things in health-care sector: An analytical approach. In *Advancement of Machine Intelligence in Interactive Medical Image Analysis*; Springer: Singapore, 2020; pp. 307–329.
125. Haque, M.A.; Haque, S.; Kumar, K.; Singh, N.K. A Comprehensive Study of Cyber Security Attacks, Classification, and Countermeasures in the Internet of Things. In *Handbook of Research on Digital Transformation and Challenges to Data Security and Privacy*; Anunciação, P.F., Pessoa, C.R.M., Jamil, G.L., Eds.; IGI Global: Hershey, PA, USA, 2021; pp. 63–90.
126. Yousuf, O.; Mir, R.N. A survey on the Internet of Things security. *Inf. Comput. Secur.* **2019**, *27*, 292–323. [CrossRef]
127. Litoussi, M.; Kannouf, N.; El Makkaoui, K.; Ezzati, A.; Fartitchou, M. IoT security: Challenges and countermeasures. *Procedia Comput. Sci.* **2020**, *177*, 503–508. [CrossRef]
128. Bagga, M.; Thakral, P.; Bagga, T. A Study on IoT: Model, Communication Protocols, Security Hazards & Countermeasures. In Proceedings of the 2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC), Solan, India, 20–22 December 2018; pp. 591–598.
129. Chaudhary, J.; Prasad, S. IoT in healthcare sector-a comprehensive analysis of threats and privacy issues. *AIP Conf. Proc.* **2022**, *2519*, 030058. [CrossRef]
130. Akhtar, M.S.; Feng, T. A Systemic Security and Privacy Review: Attacks and Prevention Mechanisms over IOT Layers. *EAI Endorsed Trans. Secur. Saf.* **2022**, *8*, e5. [CrossRef]

131. Sabir, M.A.; Sajid, A.; Ashraf, F. A Systemic Security and Privacy Review: Attacks and Prevention Mechanisms Over IoT Layers. In *Big Data Analytics and Computational Intelligence for Cybersecurity*; Ouaissa, M., Boulouard, Z., Ouaissa, M., Khan, I.U., Kaosar, M., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 69–89.
132. Gautam, S.; Malik, A.; Singh, N.; Kumar, S. Recent Advances and Countermeasures Against Various Attacks in IoT Environment. In Proceedings of the 2019 2nd International Conference on Signal Processing and Communication (ICSPC), Coimbatore, India, 29–30 March 2019; pp. 315–319.
133. Bahalul Haque, A.K.M.; Bhushan, B.; Nawar, A.; Talha, K.R.; Ayesha, S.J. Attacks and Countermeasures in IoT Based Smart Healthcare Applications. In *Recent Advances in Internet of Things and Machine Learning: Real-World Applications*; Balas, V.E., Solanki, V.K., Kumar, R., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 67–90.
134. Research, Z. Top Cyber Threats to Healthcare Organizations. Available online: https://www.zerofox.com/blog/cyber-threats-to-healthcare-industry/ (accessed on 10 December 2023).
135. Samaila, M.G.; Sequeiros, J.B.F.; Freire, M.M.; Inácio, P.R.M. Security Threats and Possible Countermeasures in IoT Applications Covering Different Industry Domains. In Proceedings of the 13th International Conference on Availability, Reliability and Security, Hamburg, Germany, 27–30 August 2018; p. 16.
136. Mehra, D.; Sharma, D.K.; Sharma, S.K. Challenges and vulnerabilities of WSN-based IoT in the healthcare and medical industry. In *Integration of WSNs into Internet of Things*; CRC Press: Boca Raton, FL, USA, 2021; pp. 305–333.
137. Fasunlade, O.; Zhou, S.; Sanders, D. Security Threats and Possible Countermeasure In Digital Healthcare. In Proceedings of the 2021 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 15–17 December 2021; pp. 1297–1302.
138. Saba, T.; Haseeb, K.; Ahmed, I.; Rehman, A. Secure and energy-efficient framework using Internet of Medical Things for e-healthcare. *J. Infect. Public Health* **2020**, *13*, 1567–1575. [CrossRef]
139. Alsaidi, A.; Kausar, F. Security Attacks and Countermeasures on Cloud Assisted IoT Applications. In Proceedings of the 2018 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, USA, 21–23 September 2018; pp. 213–217.
140. Karunarathne, S.M.; Saxena, N.; Khan, M.K. Security and Privacy in IoT Smart Healthcare. *IEEE Internet Comput.* **2021**, *25*, 37–48. [CrossRef]
141. Qadri, Y.A.; Ali, R.; Musaddiq, A.; Al-Turjman, F.; Kim, D.W.; Kim, S.W. The limitations in the state-of-the-art counter-measures against the security threats in H-IoT. *Clust. Comput.* **2020**, *23*, 2047–2065. [CrossRef]
142. Cilliers, L. Wearable devices in healthcare: Privacy and information security issues. *Health Inf. Manag. J.* **2019**, *49*, 150–156. [CrossRef] [PubMed]
143. Ianculescu, M.; Coardoș, D.; Bica, O.; Vevera, V. Security and Privacy Risks for Remote Healthcare Monitoring Systems. In Proceedings of the 2020 International Conference on e-Health and Bioengineering (EHB), Iasi, Romania, 29–30 October 2020; pp. 1–4.
144. Tyagi, A.K.; Nair, M.M. Internet of Everything (IoE) and Internet of Things (IoTs): Threat Analyses, Possible Opportunities for Future. *J. Inf. Assur. Secur.* **2020**, *15*, 153–177.
145. Aswani Devi, A.; Erukala Suresh, B.; Nayak, S.R.; Sethy, A.; Verma, A. Integrated Industrial Reference Architecture for Smart Healthcare in Internet of Things: A Systematic Investigation. *Algorithms* **2022**, *15*, 309. [CrossRef]
146. Humayun, M.; Jhanjhi, N.Z.; Alsayat, A.; Ponnusamy, V. Internet of things and ransomware: Evolution, mitigation and prevention. *Egypt. Inform. J.* **2021**, *22*, 105–117. [CrossRef]
147. Khatkar, M.; Kumar, K.; Kumar, B. An overview of distributed denial of service and internet of things in healthcare devices. In Proceedings of the 2020 Research, Innovation, Knowledge Management and Technology Application for Business Sustainability (INBUSH), Noida, India, 19–21 February 2020; pp. 44–48.
148. Mohapatro, M.; Snigdh, I. An Experimental Study of Distributed Denial of Service and Sink Hole Attacks on IoT based Healthcare Applications. *Wirel. Pers. Commun.* **2021**, *121*, 707–724. [CrossRef]
149. Bikos, A.N.; Sklavos, N. The future of privacy and trust on the internet of Things (IoT) for healthcare: Concepts, Challenges, and Security Threat Mitigations. In *Recent Advances in Security, Privacy, and Trust for Internet of Things (IoT) and Cyber-Physical Systems (CPS)*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2020; pp. 63–90.
150. Gavrilović, N.; Mishra, A. Software architecture of the internet of things (IoT) for smart city, healthcare and agriculture: Analysis and improvement directions. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 1315–1336. [CrossRef]
151. Walker-Roberts, S.; Hammoudeh, M.; Dehghantanha, A. A Systematic Review of the Availability and Efficacy of Countermeasures to Internal Threats in Healthcare Critical Infrastructure. *IEEE Access* **2018**, *6*, 25167–25177. [CrossRef]
152. Davis, J. The 10 Biggest Healthcare Data Breaches of 2019, So Far. Available online: https://healthitsecurity.com/news/the-10-biggest-healthcare-data-breaches-of-2019-so-far (accessed on 10 December 2023).
153. He, Y.; Aliyu, A.; Evans, M.; Luo, C. Health Care Cybersecurity Challenges and Solutions Under the Climate of COVID-19: Scoping Review. *J. Med. Internet Res.* **2021**, *23*, e21747. [CrossRef]
154. Burke, G.; Saxena, N. Cyber Risks Prediction and Analysis in Medical Emergency Equipment for Situational Awareness. *Sensors* **2021**, *21*, 5325. [CrossRef]
155. Irshad, R.R.; Sohail, S.S.; Hussain, S.; Madsen, D.Ø.; Zamani, A.S.; Ahmed, A.A.A.; Alattab, A.A.; Badr, M.M.; Alwayle, I.M. Towards enhancing security of IoT-Enabled healthcare system. *Heliyon* **2023**, *9*, e22336. [CrossRef] [PubMed]
156. Bygrave, L.A. Security by Design: Aspirations and Realities in a Regulatory Context. *Oslo Law Rev.* **2022**, *8*, 126–177. [CrossRef]

157. He, Y.; Huang, D.; Chen, L.; Ni, Y.; Ma, X. A Survey on Zero Trust Architecture: Challenges and Future Trends. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 6476274. [CrossRef]
158. Mishra, V.; Gupta, K.; Saxena, D.; Singh, A.K. A Global Medical Data Security and Privacy Preserving Standards Identification Framework for Electronic Healthcare Consumers. *IEEE Trans. Consum. Electron.* **2024**, *70*, 4379–4387. [CrossRef]
159. Cirne, A.; Sousa, P.R.; Resende, J.S.; Antunes, L. IoT security certifications: Challenges and potential approaches. *Comput. Secur.* **2022**, *116*, 15. [CrossRef]
160. HIPAA. Submitting Notice of a Breach to the Secretary. Available online: https://www.hhs.gov/hipaa/for-professionals/breach-notification/breach-reporting/index.html (accessed on 15 March 2024).
161. Osama, M.; Ateya, A.A.; Sayed, M.S.; Hammad, M.; Pławiak, P.; Abd El-Latif, A.A.; Elsayed, R.A. Internet of Medical Things and Healthcare 4.0: Trends, Requirements, Challenges, and Research Directions. *Sensors* **2023**, *23*, 7435. [CrossRef]
162. Cremer, F.; Sheehan, B.; Fortmann, M.; Kia, A.N.; Mullins, M.; Murphy, F.; Materne, S. Cyber risk and cybersecurity: A systematic review of data availability. *Geneva Pap. Risk Insur. Issues Pract.* **2022**, *47*, 698–736. [CrossRef]
163. Madanian, S.; Nakarada-Kordic, I.; Reay, S.; Chetty, T.h. Patients' perspectives on digital health tools. *PEC Innov.* **2023**, *2*, 100171. [CrossRef]
164. Chinbat, T.; Madanian, S.; Airehrour, D.; Hassandoust, F. Machine learning cryptography methods for IoT in healthcare. *BMC Med. Inform. Decis. Mak.* **2024**, *24*, 153. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Review

Watch the Skies: A Study on Drone Attack Vectors, Forensic Approaches, and Persisting Security Challenges

Amr Adel [1,2,*] and Tony Jan [1]

[1] Centre for Artificial Intelligence Research and Optimization (AIRO), Torrens University Australia, Ultimo, NSW 2007, Australia; tony.jan@torrens.edu.au
[2] School of Computing, Eastern Institute of Technology, Auckland 1010, New Zealand
* Correspondence: amr.adel@torrens.edu.au or avandenadel@eit.ac.nz

Abstract: In the rapidly evolving landscape of drone technology, securing unmanned aerial vehicles (UAVs) presents critical challenges and demands unique solutions. This paper offers a thorough examination of the security requirements, threat models, and solutions pertinent to UAVs, emphasizing the importance of cybersecurity and drone forensics. This research addresses the unique requirements of UAV security, outlines various threat models, and explores diverse solutions to ensure data integrity. Drone forensics, a field dedicated to the investigation of security incidents involving UAVs, has been extensively examined and demonstrates its relevance in identifying attack origins or establishing accident causes. This paper further surveys artifacts, tools, and benchmark datasets that are critical in the domain of drone forensics, providing a comprehensive view of current capabilities. Acknowledging the ongoing challenges in UAV security, particularly given the pace of technological advancement and complex operational environments, this study underscores the need for increased collaboration, updated security protocols, and comprehensive regulatory frameworks. Ultimately, this research contributes to a deeper understanding of UAV cybersecurity and aids in fostering future research into the secure and reliable operation of drones.

Keywords: unmanned aerial vehicles (UAVs); drone cybersecurity; threat models; drone forensics; security challenges; forensic methodologies

1. Introduction

Unmanned aerial vehicles (UAVs), more commonly known as drones, have seen exponential growth in their applications over the past decade. Once exclusively associated with military and defense operations, drones now permeate a plethora of sectors ranging from agriculture and surveillance to package delivery and environmental monitoring. According to the Federal Aviation Administration (FAA), the number of commercial drones in the United States is expected to have tripled by 2023, reaching an estimate of over 835,000 [1]. Similarly, the Consumer Technology Association predicts that global sales of drones will be approximately 20 million by 2034. This escalating ubiquity underscores the importance of understanding and addressing the cybersecurity and forensic aspects related to UAVs.

Drone forensics, a niche yet rapidly developing field within digital forensics, involves the process of uncovering, analyzing, and interpreting digital data from a drone system for investigative purposes. Whether it is to probe a security incident, identify an attacker, or determine the cause of an accident, drone forensics provide valuable insights into the events surrounding the UAV. This field has become more critical as drones are equipped with high-capacity storage, advanced sensors, and complex communication systems, which often hold sensitive data and are susceptible to various cybersecurity threats.

A crucial component in the UAV architecture, the ground control system (GCS), serves as the hub for human–drone interaction. It is the control station from which operators

direct the flight of a drone and manage its functionalities. While the GCS provides a means to leverage the versatility of drones, it also exposes a vulnerable attack surface for cybercriminals. The vulnerabilities of the GCS and their implications for drone operation and security form an essential aspect of UAV security and forensics.

The significance of drone forensics and security cannot be overstated in today's increasingly drone-dependent world. As drones continue to find applications in security-sensitive sectors, such as military surveillance, law enforcement, and delivery of goods, any security lapses could result in dire consequences, including threats to national security, invasion of personal privacy, and loss of valuable assets. This underscores the urgent need for robust cybersecurity measures and effective forensic methodologies that can secure UAVs against potential attacks and aid in the investigation post any security incidents.

This paper used a systematic literature review to understand the field comprehensively. We searched Scopus and IEEE Xplore with keywords: "Unmanned Aerial Vehicles (UAVs)", "Drone Cybersecurity", "Drone Cybersecurity Threat Models", "Drone Forensics", "Security", and "Drone Forensic Methodologies" for articles published since 2018. By aggregating and analyzing scholarly articles, industry reports, case studies, and government publications, this paper presents a holistic view of current drone threat models, forensic approaches, and security challenges.

1.1. Research Methodology and Selection Criteria

The research methodology of this review includes a systematic analysis of the existing literature, case studies, and empirical data on drone security, forensic approaches, and persistent security challenges. This approach aims to provide a comprehensive understanding of UAV security and forensics, highlighting key components, challenges, and potential solutions. The research methodology and source selection criteria are outlined below:

A thorough search of scholarly databases, online repositories, and professional journals was conducted to find relevant articles, books, reports, and conference proceedings on drone security, forensics, and industry applications. A systematic search query using relevant keywords and phrases related to UAV security, forensics, and industry applications was designed. Data were gathered from various platforms, including Google Scholar, ACM Digital Library, ScienceDirect, IEEE Xplore, Scopus, and Springer using the following search queries:

1. Query A: (("drone security" OR "UAV security" OR "unmanned aerial vehicle security") AND ("forensics" OR "cybersecurity" OR "digital forensics") AND ("smart cities" OR "urban management"));
2. Query B: (("drone forensics" OR "UAV forensics") AND ("data integrity" OR "digital evidence" OR "incident response") AND ("cyber attacks" OR "security threats"));
3. Query C: (("unmanned aerial vehicle" OR "drone technology") AND ("threat models" OR "attack vectors" OR "security vulnerabilities") AND ("data protection" OR "encryption" OR "secure communication"));
4. Query D: (("drone operation" OR "UAV deployment") AND ("security protocols" OR "forensic methodologies") AND ("case studies" OR "real-world applications"));
5. Query E: (("drone forensic analysis" OR "UAV forensic techniques") AND ("machine learning" OR "artificial intelligence" OR "predictive analytics") AND ("emerging trends" OR "future research" OR "innovation")).

The research selection process, detailed in Appendix A, began with 2154 records identified through database searches. After removing 1608 duplicates and ineligible records, 546 remained for screening. Of these, 383 were excluded based on relevance and quality. Out of 163 papers sought, 47 were not retrievable, leaving 116 for eligibility assessment. Exclusions included 40 non-peer-reviewed, 25 irrelevant, 10 lacking empirical support, 17 older publications, and 11 non-English papers, totaling 103 exclusions. Ultimately, 13 studies were included in the review, with 113 studies considered for final analysis.

The review followed strict inclusion and exclusion criteria to ensure study relevance and quality, as detailed in Appendix B. The inclusion criteria were: peer-reviewed status,

relevance to UAV security and forensics, empirical evidence, publication within the last 10 years, and availability in English. Each study was verified for these criteria. Exclusions were non-peer-reviewed, irrelevant, lacking empirical support, older than 10 years, or not in English. This systematic approach ensured the review included only high-quality, relevant, and recent studies, enhancing the robustness and reliability of the findings.

1.2. Comparison to Other Survey Papers and Contributions

Several surveys have explored the landscape of UAV security and forensics, each focusing on different subtopics. In 2023, Sighag et al. [2] presented a Cyber4Drone survey paper that provided a systematic review of cybersecurity and forensic challenges for unmanned aerial vehicles (UAVs), including an analysis of threat models, security and privacy issues, and a taxonomy of drone forensic techniques and tools. In the same year, Ceviz et al. [3] published a survey paper focusing on security issues, threats, and solutions for unmanned aerial vehicles (UAVs) and flying ad hoc networks (FANETs), including an analysis of the attack surface, potential threats, prevention and detection countermeasures, and simulations of four routing attacks on FANETs. Shafik et al. [4] provided an overview of the cybersecurity threats and challenges faced by unmanned aerial vehicles (UAVs), explored various cybersecurity strategies and techniques to protect UAVs, and identified future research directions to mitigate cybersecurity risks in UAVs. In 2022, Pandey et al. [5] conducted a comprehensive survey on security threats and mitigation techniques in UAV communications. Siddiqi et al. [6] presented an analysis of the security- and privacy-related concerns of UAVs, offering countermeasures and recommendations. Wang et al. [7] provided a comprehensive survey of the current achievements in physical layer security (PLS) for UAV communications.

Despite these valuable contributions, several gaps remain in the existing literature. Many surveys have focused on specific aspects of UAV security or forensics and often lack a comprehensive analysis. Our study aims to provide a holistic overview encompassing security requirements, threat models, and forensic approaches. The rapid evolution of UAV technology has introduced new security challenges that necessitate ongoing research. Existing surveys often lack a forward-looking perspective to address emerging threats. Thus, there is a critical need for a unified framework that integrates cybersecurity practices with forensic methodologies to enhance both preventive and investigative capabilities.

Our survey significantly advanced the field of UAV security and forensics by providing an integrated analysis that bridges cybersecurity measures with forensic methodologies. This comprehensive approach covers security requirements, threat models, attack vectors, forensic methodologies, and mitigation techniques. It identifies emerging challenges due to the rapid evolution of UAV technology, proposes future research directions, and emphasizes the need for continuous innovation. The survey advocates for a unified framework to enhance both preventive and investigative capabilities and supports standardized protocols and best practices. By synthesizing previous research, it highlights key contributions and gaps and offers practical recommendations for improved UAV security and forensics. Serving as a crucial resource, it supports ongoing research, industry applications, and policy development across interdisciplinary fields, ensuring that the findings are broadly relevant and impactful.

The contribution of this paper lies in its exhaustive review of drone threat models, exploration of forensic approaches used for investigating drone-related incidents, and analysis of persistent security challenges. In addition, it delivers a systematic taxonomy of drone forensics, detailing key artifacts, relevant tools, and benchmark datasets. This survey will serve as a significant resource for researchers, practitioners, and policymakers aiming to bolster the security and integrity of UAV operation.

The remainder of this paper is organized as follows: Section 2 provides an overview of drone technologies. Section 3 addresses drone security vulnerabilities and evolving attack vectors. Section 4 focuses on a detailed review of drone security solutions. Section 5 presents intriguing future research opportunities, and Section 6 concludes the paper.

2. Overview of Drone Technologies

2.1. Drone Anatomy

The anatomy of a drone is a sophisticated integration of various technological components that enables it to perform a wide range of functions, from simple recreational flying to complex commercial tasks [8].

Figure 1 illustrates the essential components and functionalities of drones, including the flight controller, propellers, brushless motors, and gimbal systems, and their roles in operation. Each part is vital for the drone's flight, navigation, and tasks like video recording or streaming [9].

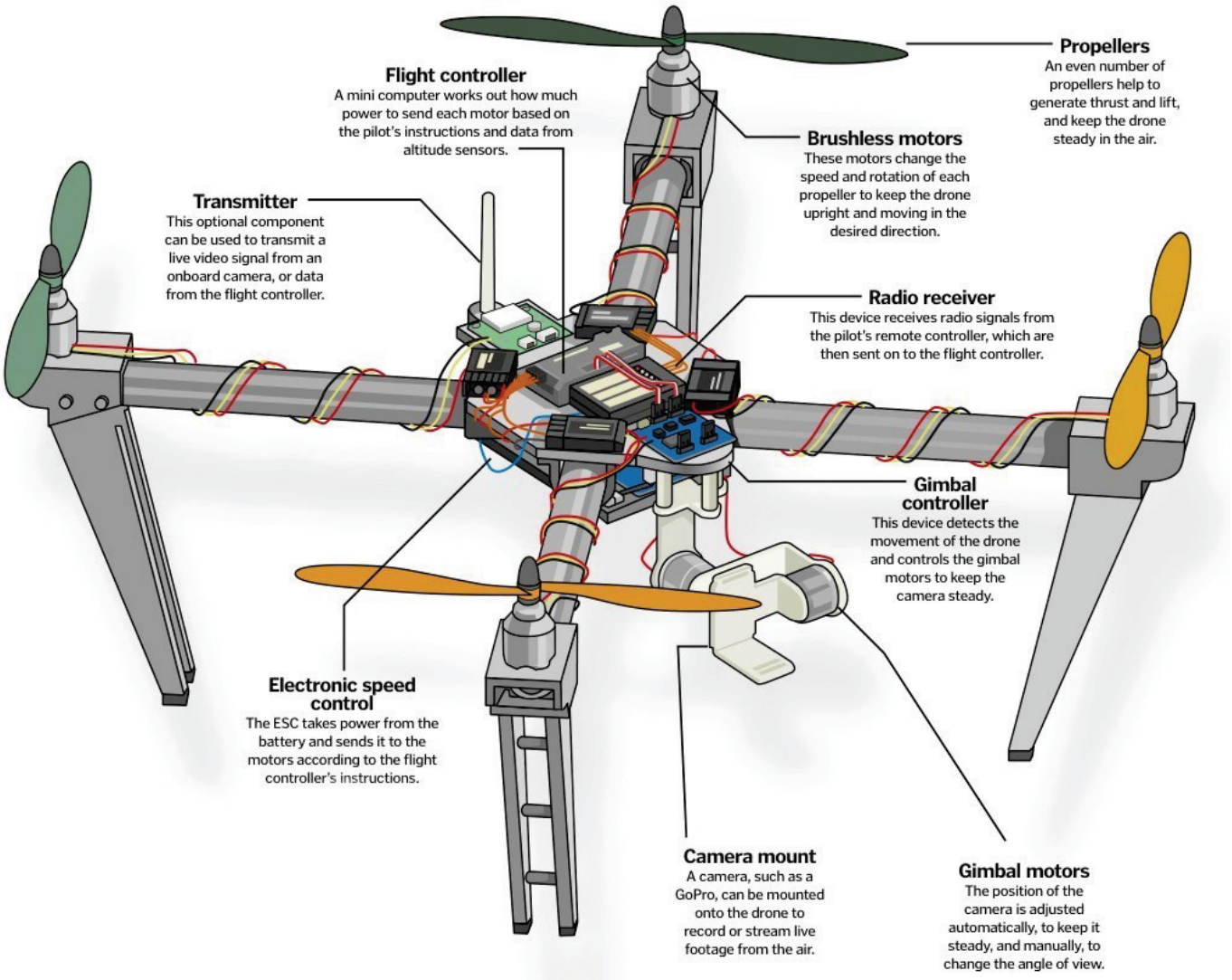


Figure 1. Anatomy of drones.

The flight controller, a minicomputer, processes sensor inputs and pilot commands to maintain stable flight, enabling both basic maneuvers and advanced navigation. Brushless motors, chosen for efficiency and reliability, adjust propeller speed and rotation to create lift and thrust, which are crucial for speed and stability [10]. Propellers are aerodynamically designed to optimize performance.

Communication with the operator is managed through a radio receiver and transmitter, which handle commands and data such as live video feeds or telemetry [11]. The gimbal controller and motors stabilize mounted devices like cameras, ensuring steady

footage. Electronic speed controllers (ESCs) regulate motor power from the battery to adjust propeller speed for smooth flight dynamics [12].

Each drone component enhances its flight capabilities and broadens its applications, making drones versatile tools for civilian and commercial use for usages such as aerial photography or complex surveillance [13].

Drone operations involve multiple layers to ensure optimal performance across tasks, with usages ranging from aerial photography to complex data collection and analysis [14–17]. The architecture of a drone system is divided into specific functional layers:

- **Planning layer**: Defines missions or services and develops executable plans; this is essential for disaster management.
- **Flight management layer**: Executes the planned route, managing dynamic flight, obstacle avoidance, and real-time path modifications to ensure secure mission completion.
- **Control layer**: Interfaces directly with drone hardware, sending commands to sensors and actuators and handling real-time adjustments to maintain the stability and trajectory.

The drone's hardware includes the frame, motors, propellers, and electronic speed controllers (ESCs) and enables physical operation. Sophisticated equipment like GPS and antenna systems ensure communication with ground control and navigational accuracy via GPS and RTK networks. In scenarios with multiple drones, robust data link systems facilitate drone-to-drone communication for coordinated task execution [18,19]. Figure 2 illustrates the multi-layered approach to drone operations and highlights key components and communication frameworks that enable autonomous and coordinated functions.

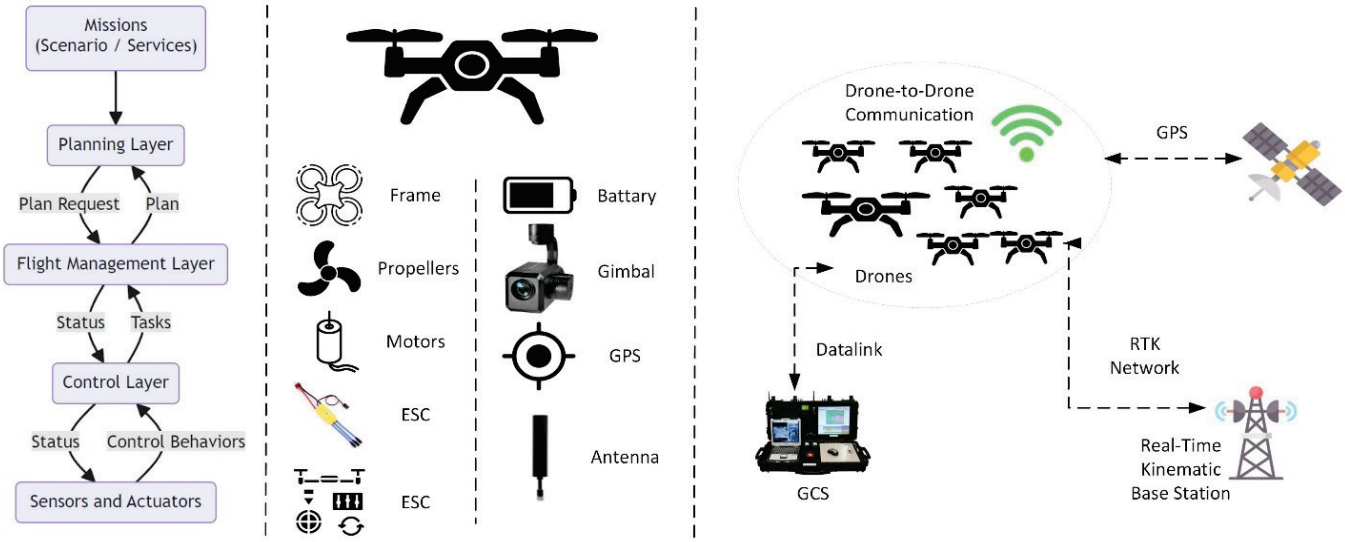


Figure 2. Drone operations and system architecture.

2.2. Drone Forensics Artifacts

The meticulous examination and analysis of various artifacts are paramount to understanding the operations and interactions of these aerial devices within their operational environments. These artifacts, which range from flight logs to system firmware, provide a rich tapestry of data critical for forensic investigators [20–24]. Each artifact holds specific insights into the drone's performance, activities, and any anomalies, which may indicate misuse or malfunctions.

Table 1 categorizes key drone forensic artifacts and details their primary sources and descriptions.

Table 1. Drone forensic artifacts: E is for Exif data, F is for system files, G is for ground controller, L is for logs, M is for memory card, and O is for observation.

Artifact Name	Source	Description
Flight Logs	L	Records of flight data including times, altitudes, and GPS coordinates.
System Firmware	F	The embedded software that controls drone operations.
GPS Data	E, L	Data capturing the drone's geographical position during flights.
Controller Inputs	G	Records of commands input by the operator during flight.
Video Files	M	Recorded footage from drone flights stored on memory cards.
Photo Files	M, E	Images captured during flight, often containing Exif metadata such as timestamps and camera settings.
Battery Information	L	Data regarding battery status and history during flights.
Communication Logs	L, G	Logs detailing the communication between the drone and its ground controller.
Error Reports	F, L	System-generated reports detailing malfunctions or errors during operation.
Maintenance Records	F, L	Logs related to drone servicing, updates, and repairs.
Wi-Fi Data	L	Information about Wi-Fi networks used for control and data transmission.
Serial Number	F	Unique identifier of the drone, often embedded in system files or visible on the drone body.
Telemetry Data	L	Real-time data on various flight parameters such as speed, altitude, and orientation.
Crash Reports	L	Detailed reports generated when a drone experiences a crash or significant malfunction.
Configuration Files	F	Files that determine the settings and options of the drone's operating system.
Propeller Data	O	Observations and data regarding the condition and performance of the drone's propellers.
Firmware Update Logs	F	Logs documenting the history and details of firmware updates applied to the drone.
Environmental Data	L	Data collected during flight related to environmental conditions such as temperature and wind speed.
Security Protocols	F	Information regarding the encryption and security measures used to protect drone communications and data storage.

3. Drone Vulnerabilities and Attack Vectors

The emerging risks associated with drone technology pose significant threats to data security, infrastructure integrity, and public safety [25]. Attackers may exploit zero-day vulnerabilities and security loopholes to infiltrate drone communication networks [26,27]. Drone forensics has emerged as a critical field that aims to determine attackers' motives through meticulous investigations [28]. This systematic process involves the collection, preservation, and analysis of digital, software, and hardware evidence from drones [29]. Furthermore, drone forensics can guide the development of new technologies or policies that mitigate the effects of similar future attacks and enhance overall security levels [30].

This section explores various drone security attacks to provide structured guidance for the forensic investigation of drones. Vulnerabilities in communication media and frequency-based weaknesses compound these risks. Even advanced drones that are equipped with cameras and GPS capabilities are not immune to such threats. A taxonomic classification of drone attacks detailing their impacts, execution tools, and mechanisms is presented in Table 2.

Table 2. Categorization of drone attack methods for target zones (Z). Z1: UAVs, Z2: communication systems, Z3: control hubs, Z4: command centers, and Z5: Regulatory Bodies, Production Facilities, and Associated Equipment.

Drone Attacks	Tools/ Mechanisms	Impact	Security Requirements	Attack Surfaces	Key Papers
GPS Spoofing	GPS signal simulators	Misdirection, route deviation	Enhanced route security	Z1: UAVs	[31–34]
Signal Jamming	Radio frequency jammers	Loss of control, crashing	Robust signal integrity	Z1: UAVs, Z2: Communication Systems	[35–39]
Unauthorized Access	Hacking tools	Data theft, control takeover	Access control improvements	Z3: control hubs, Z4: command centers	[40–45]
Physical Attack	High-energy lasers	Damage, destruction	Structural integrity checks	Z1: UAVs	[46–50]
Network Intrusion	Malware, spyware	Data breach, system compromise	Enhanced cybersecurity measures	Z2: Communication Systems, Z4: command centers	[51–54]
Battery Tampering	Physical interference	Power loss, mid-air failure	Reliable power supply systems	Z1: UAVs	[55,56]
Firmware Hacking	Custom firmware	Altered behavior, backdoors	Secure firmware protocols	Z1: UAVs, Z3: Control Hubs	[57–63]
Sensor Blinding	Directed bright lights	Impaired vision, collision	Improved sensor protection	Z1: UAVs	[64–68]
Denial of Service	Flooding networks	Disrupted operations	Network resilience	Z2: Communication Systems, Z3: Control Hubs	[69–74]
Data Interception	Sniffing tools	Espionage, data leakage	Data encryption standards	Z2: Communication Systems	[75–79]
Protocol Exploitation	Exploitation kits	Command hijacking	Secure communication protocols	Z3: Control Hubs	[80–86]
Malicious Code Injection	Trojans, viruses	Malfunctions, unsafe operations	Malware detection systems	Z1: UAVs, Z4: command centers	[87,88]
Ransomware Attack	Ransomware	Locked systems, ransom demand	Anti-ransomware strategies	Z4: command centers	[4,89–91]
Eavesdropping	Audio–visual surveillance	Privacy invasion	Privacy safeguards	Z2: Communication Systems, Z4: command centers	[92–95]
Supply Chain Attack	Compromised components	Integrated vulnerabilities	Supply chain security	Z5: regulatory bodies, production facilities, and associated equipment	[96–98]
Insider Threat	Sabotage by insiders	System sabotage, data theft	Internal security measures	Z3: Control Hubs, Z4: command centers	[99–102]
Regulatory Non-compliance	Bypassing controls	Legal penalties, shutdown	Compliance management	Z5: regulatory bodies, production facilities, and associated equipment	[103–105]

3.1. GPS Spoofing

GPS spoofing transmits false GPS signals to manipulate a drone's navigation to redirect it from its intended path. This can lead to theft, operational interference, or unauthorized payload delivery and poses serious risks in security-sensitive environments. Defenses

include advanced cryptographic techniques, multi-sensor fusion, and anomaly detection systems to identify and mitigate deviations [106,107].

3.2. Signal Jamming

Signal jamming disrupts drone communication by flooding frequencies with high-intensity signals, severing the link between the drone and its operator. This can cause loss of control, autonomous landing, or system failure, causing risk to critical applications like law enforcement or emergency response. Countermeasures involve frequency-hopping and spread-spectrum technologies and robust fail-safe mechanisms to guide drones to safety during communication loss [108].

3.3. Network Intrusion

Network intrusion involves hacking drone control channels to intercept data or insert malicious commands, leading to data theft, surveillance, payload hijacking, or weaponization. This threat is heightened with IoT-connected drones. Protection requires secure communication protocols, regular software updates, and comprehensive network monitoring to detect and respond to unauthorized access [109].

3.4. Malicious Code Injection

Malicious code injection targets drone software through compromised firmware updates or security vulnerabilities. This can manipulate drone behavior, disable safety features, or extract confidential data, making drones unreliable or dangerous. Prevention includes strict code integrity checks, secure update mechanisms, and regular security audits [110].

3.5. Physical Tampering

Physical tampering involves altering drone hardware components like microSD cards, sensors, or communication modules, potentially enabling espionage or sabotage. Defending against this requires tamper-detection systems and strict physical security protocols, such as utilizing physical unclonable functions (PUFs) [111].

3.6. Eavesdropping

Drones equipped with surveillance tools can eavesdrop, gathering sensitive information undetected. This poses privacy risks and potential corporate espionage threats. Countermeasures include no-fly zones, anti-drone technologies, and strict regulatory measures to prevent unauthorized drone flights over private property [112].

3.7. Supply Chain Interference

Supply chain interference involves inserting vulnerabilities during drone manufacturing, which would affect entire fleets if the compromised components are widespread. Mitigation requires stringent security protocols, vetting suppliers, and thorough security audits and tests on components before integration [113].

4. Drone Forensics and Security Solutions—Review

In the realm of drone cybersecurity and forensics, a number of studies have delved into identifying threat models, evaluating security measures, and developing forensic methodologies.

In 2017, Renduchintala et al. [114] made substantial strides in the realm of drone forensics by proposing a comprehensive framework for the examination of digital flight logs from micro-drones. The authors recognized the security concerns arising from the increased public use of unmanned aerial vehicles (UAVs), or drones. Given the drones' ability to access potential targets closely, the authors were motivated to develop a forensic framework capable of thoroughly inspecting the drone's activities post-flight. Specifically, their paper scrutinized the crucial log parameters of an autonomous drone and proposed an in-depth software architecture related to drone forensics. Preliminary results indicated

the potential of their under-development software, which boasted a user-friendly graphical user interface (GUI) that allowed users to extract and examine onboard flight information. Their work is significant because it aims to equip the forensic science community with a practical tool to investigate drone-related crime cases more effectively.

In 2018, Azhar et al. [115] tackled the growing need for effective forensic analysis of drones, particularly those involved in unlawful activities. They recognized the substantial rise in drone-related offenses due to their easy accessibility and robust carrying capacities. To address this, the authors focused on extracting and identifying significant artifacts from recorded flight data and associated mobile devices. They employed a range of open-source tools, which were complemented by basic scripts they developed to facilitate the analysis. Their research specifically examined two popular drone systems: the DJI Phantom 3 Professional and the Parrot AR Drone 2.0. The authors stressed the importance of adhering to forensically sound methods as per the guidelines of the Association of Chief Police Officers (ACPO). Their findings emphasized that, while drone operations vary, generic methods can be effectively applied for the extraction and analysis of data from drones and associated devices, thereby significantly contributing to advancements in the drone forensics field.

In the same year, Bouafif et al. [116] addressed the field of drone forensics and the associated challenges. They emphasized the growing popularity of unmanned aerial vehicles (UAVs), or drones, which possess powerful information acquisition and processing capabilities along with intelligent surveillance and reconnaissance features. However, the misuse of drones for illegal and potentially criminal activities poses significant threats to individuals, organizations, public safety, and national security. Despite the increasing importance of drone forensics, it remains a relatively underexplored research topic. The authors presented important results from a forensic investigation analysis conducted on the Parrot AR Drone 2.0 and offered new insights into the field.

In 2019, Iqbal et al. [117] explored drone forensics, focusing on identifying and analyzing drone vulnerabilities. They highlighted the susceptibility of drones to various types of attacks, such as GPS spoofing and de-authentication, which pose challenges to forensic investigations and could potentially lead to criminal activities. The authors advocated standardized drone forensics to enhance security, identify vulnerabilities, and resolve drone-related crimes. Their research included a case study of potential attacks on the Parrot Bebop 2 drone and the process of evidence collection. They also proposed a framework for small-scale drone forensics and drone ontology for context data modeling. Their work provided significant insights into drone vulnerabilities and the importance of forensic science in managing drone-related crimes.

In the same year, Kao et al. [118] focused on drone forensic investigation, with the DJI Spark drone serving as a case study. The authors highlighted the rise in drone crimes due to the carrying capabilities and widespread availability of drones. They emphasized the importance of drone forensic analysis in addressing these crimes. The paper detailed the process of collecting, examining, correcting, and analyzing crucial artifacts extracted from the recorded flight data. It also presented crime reconstruction techniques for temporal analysis and explored the associations between drones, mobile phones, and SD cards. By exploring and evaluating these relational artifacts, the authors aimed to uncover valuable insights into the relationships among the different components involved in drone-related crimes. This work contributes to the field of drone forensics by providing a systematic approach for artifact analysis and crime reconstruction, focusing on the DJI Spark drone.

In 2019, Mantas et al. [119] addressed the field of drone forensics, with a specific focus on forensics related to flight data logs. They recognized the decreasing cost of unmanned aerial vehicles (UAVs) and drones, which has led to their widespread adoption in various civilian and business applications. However, the features offered by drones have also been maliciously exploited, thereby increasing the need for drone forensics. This work aims to fill a gap in the literature by investigating forensics on flight data logs, specifically focusing on the widely used Ardupilot platform and its dataflash and telemetry logs. The

authors discussed a methodology for collecting the necessary information, analyzing it, and constructing a timeline of events. They also developed an open-source tool that facilitates this process and tested it using data provided by VTO Labs. This study contributes to the field of drone forensics by providing insights into the analysis of flight data logs and offering a practical tool for forensic investigations in this domain.

In 2020, Yousef et al. [120] delved into drone forensics by performing a meticulous analysis of the emerging DJI models. They acknowledged the challenges posed by rapidly evolving drone technology, particularly due to drones' escalating use in digital crimes. The authors emphasized the need for forensic examiners to be well-versed in drone technology, forensic methods, and the capabilities of existing tools to effectively extract crucial information for forensic investigations. Their study analyzed the data extracted from four popular hobbyist drone models (DJI Mavic 2 Pro, DJI Mavic Air, DJI Spark, and DJI Phantom 4) and compared the applicability and capability of several commercial and open-source forensic tools. Their findings highlighted the difficulties in analyzing newer drone models due to enhanced security measures. Therefore, they stressed the need for novel forensic processes and specialized tools to enhance drone forensic analysis.

In 2021, Al et al. [121] conducted a case study to explore drone forensics and digital forensic investigations for common drone models. The authors highlighted the growing prevalence of drones, which have become a societal norm in our daily lives. They emphasized the privacy challenges posed by drones, as they can capture high-quality aerial photos, store and transmit data, and potentially invade privacy or expose sensitive information if misused or intercepted by hackers. Recognizing the increasing misuse of drones in criminal activities, the authors stressed the need for a novel methodological approach to conduct digital forensic analyses on seized drones. This study investigated six popular drone brands that are commonly associated with criminal activities and extracted forensically relevant data such as location information, captured images and videos, flight paths, and ownership details of confiscated drones. The experimental results demonstrated the potential of drone forensics in aiding law enforcement agencies by providing valuable information that is crucial for criminal investigations.

In the same year in 2021, Al et. al., [122] addressed the opportunities in drone forensic models. They acknowledged the significant impact of unmanned aerial vehicles (UAVs) or drones on surveillance and supply chain logistics by enabling access to previously inaccessible areas. However, the increased adoption of drones has also led to an upsurge in drone-related criminal activities, raising concerns related to security and forensics. To gain a deeper understanding of the current state of research and potential mitigation approaches, the authors conducted a detailed review of existing digital forensic models using a Design Science Research method. This study provides valuable insights into research challenges and opportunities in investigating drone-related incidents. The authors propose a potential generic investigation model and emphasize the relevance of these findings for forensic researchers and practitioners for developing a guided methodology for drone-related event investigation. Additionally, they highlighted the importance of this study as a foundation for the development of international standardization for drone forensics.

In their seminal work in 2022, Alotaibi et. al. [123] proposed a novel framework aimed at improving the readiness of forensic investigations in the drone field. Recognizing the unique complexities and challenges associated with drone forensics, the authors proposed a comprehensive model that emphasized preparedness to efficiently collect, analyze, and preserve the data involved in drone-related incidents. Their framework offers a structured approach to digital evidence handling, which is key to successful forensic investigations. Importantly, it is designed to be applicable in diverse scenarios, regardless of the specificities of the drone system or nature of the incident. This contribution is significant because it enhances the capabilities of forensic teams for dealing with drone-related security breaches, thereby improving the overall security landscape of the rapidly evolving UAV sector.

In 2022, Lan et al. [124] focused on drone forensics: specifically, by conducting a case study of the DJI Mavic Air 2 drone. They acknowledged the increasing prevalence of cost-

effective and high-performance unmanned aerial vehicles (UAVs) in the consumer market, which leads to an increase in their leisure and business applications. Consequently, there has been growing demand for digital forensic examinations of these devices. The authors explored and discussed the forensic examination process specifically for DJI drones, which are popular in Singapore. They presented their findings by examining the exposed File Transfer Protocol (FTP) channel and extracting the data-at-rest from the drone's memory chip. The extraction was carried out using the chip-off and chip-on techniques, and the authors demonstrated their methodology for retrieving data from the drone. This research contributes to the field of drone forensics by providing insights into the forensic examination process for DJI drones, thus contributing to our understanding of the digital forensics of UAVs.

In 2022, Da Silva et al. [125] developed an efficient platform using the MQTT protocol for UAV control and DoS attack detection and demonstrated its robustness through tests on latency and network and memory consumption with a high true positive rate for DoS detection using advanced machine learning techniques. In 2023, Buccafurri et al. [126] addressed the need for communication anonymity within MQTT for IoT networks by proposing MQTT-anonymous (MQTT-A), which ensured the anonymity of publishers and subscribers through P2P collaboration among intermediate bridge brokers by using standard MQTT primitives and without requiring changes to existing infrastructure. This protocol was validated through global experimental tests and showed reasonable latency tradeoffs and minimal impact on network performance. In 2019, Xiong et al. [127] introduced a hybrid SDN and MQTT communication system for battlefield UAV swarms that was designed to be flexible, adaptable, and intelligent in order to support dynamic swarm formations, flexible data transmission, and enhanced security with a QoS-based multi-path routing framework. Case studies validated its practical effectiveness, although a more detailed discussion of the challenges and limitations would strengthen these studies. Collectively, these papers make significant contributions to secure, efficient, and anonymous communication systems for UAVs.

In 2022, Baig et al. [128] focused on drone forensics and machine learning with the aim of sustaining the investigation process. They recognized the increasing adoption of drones to address various challenges and provide support and convenience in areas such as medical supply delivery, surveillance, weather data collection, and home delivery services. However, the authors also acknowledge that drones have been misused for criminal activities. Their research entails a survey of artificial intelligence techniques in the literature that are relevant for processing drone data and uncovering criminal activity. Additionally, they proposed a novel model that leveraged machine learning to classify drone data as part of a digital forensic investigation. The authors concluded that properly trained machine learning models hold promise for accurately assessing drone data obtained from crime scenes. Their work paves the way for academia and industry practitioners to adopt machine learning techniques for the effective use of drone data in forensic investigations, contributing to the advancement of drone forensics.

In more recent years, the following key technologies have emerged in drone security:

Drone security module: The work of Schiller [129] in 2023 and the earlier work of Kim et al. [130] considered a drone security module that can encrypt control signals and telemetry data between the UAV and the ground control station (GCS). These encryption mechanisms employed advanced cryptographic algorithms to secure data in transit, protecting against eavesdropping and unauthorized access. Their modules ensured the confidentiality (C) and integrity (I) of the communications, targeting UAVs (Z1), communication systems (Z2), and control hubs (Z3). However, their modules did not specifically address the availability (A) of the system, leaving it susceptible to denial-of-service (DoS) attacks. A further review of drone security modules was discussed by Samantha et al. [131].

Blockchain for secure data storage: The work of Cheema et al. [132], Bera et al. [133], Singh et al. [134], and Gupta et al. [135] considered blockchain technology to secure data transmitted to and from drones. By creating an immutable ledger of all interactions, the

blockchain ensured that data integrity was maintained, preventing tampering or unauthorized modifications. These solutions targeted all security zones, including UAVs (Z1), communication systems (Z2), control hubs (Z3), command centers (Z4), and regulatory bodies (Z5). It provided comprehensive security coverage across confidentiality (C), integrity (I), and availability (A), although the potential for signal hijacking remained a concern.

Drone-assisted public safety networks: The work of Minhas et al. [136] and He et al. [137] proposed that UAVs augment public safety networks by being deployed to strategic positions. Their approaches utilized real-time data transmission and processing to enhance situational awareness and response times. The primary security targets were UAVs (Z1), communication systems (Z2), control hubs (Z3), and command centers (Z4). Their solutions enhanced integrity (I) and availability (A) by ensuring reliable data transmission and operational continuity in public safety operations. A further review of drone-assisted public safety networks is shared by Ali et al. [138].

Machine learning for threat detection: Studiawan et al. [139] in 2023, Abu et al. [140], and Guerber et al. [141], alongside many scholars, studied various machine learning algorithms, such as the random forest classifier, to be implemented to detect network attacks, including DoS, port scanning, and brute force attempts. This predictive analytics approach analyzed network traffic patterns to identify anomalies indicative of security breaches. It targeted UAVs (Z1), communication systems (Z2), control hubs (Z3), command centers (Z4), and regulatory bodies (Z5). The solution enhanced confidentiality (C), integrity (I), and availability (A) by providing the early detection and mitigation of threats. A comprehensive review was provided by Heidari et al. [142] in 2023.

Blockchain for UAV signal security: The work of Haefeez et al. [143] in 2023, Kumar et al. [144], Ch et al. [145], and Rana et al. [146] applied blockchain technology to secure the signal transmission between controllers and UAVs. Their methods secured the communication protocol, ensuring that signals were authentic and had not been tampered with. The primary targets were UAVs (Z1), communication systems (Z2), control hubs (Z3), and command centers (Z4). While enhancing confidentiality (C) and integrity (I), the approach required robust mechanisms to ensure signal availability (A).

Software-defined networks (SDNs) for drone swarms: The works of Agnew et al. [147] in 2024, Guerber et al. [141] considered SDN solutions to manage and coordinate drone swarms, facilitating centralized control and dynamic responses to security threats. Their architectures separated the control plane from the data plane, allowing for more flexible and responsive security management. Their target areas included UAVs (Z1), communication systems (Z2), and control hubs (Z3). An SDN enhances confidentiality (C) and integrity (I), but continuous monitoring is necessary to maintain availability (A) against coordinated attacks.

Light-weight hardware security: Pu et al. [148] in 2024 and [149] in 2023 considered light-weight hardware solutions to ensure the confidentiality and integrity of command data and payload data. These hardware-based security measures were integrated directly into the drone's components, providing a tamper-resistant environment. The primary targets were UAVs (Z1), communication systems (Z2), and control hubs (Z3). These solutions significantly enhanced confidentiality (C) and integrity (I) but may not fully address availability (A) concerns.

Multi-sensor detection systems: The work of Famili et al [150] considered multi-sensor detection systems using various sensors to detect unauthorized drone activity in restricted areas. This approach enhances detection accuracy by cross-referencing data from multiple sources. The key targets were communication systems (Z2), control hubs (Z3), and regulatory bodies (Z5). These systems improve integrity (I) and availability (A) by ensuring timely detection and response to intrusions.

Table 3 lists recent drone security solutions and their associated defense mechanisms.

Table 3. The state-of-the-art drone security solutions. Z1: UAVs, Z2: communication systems, Z3: control hubs, Z4: command centers, and Z5: regulatory bodies, production facilities, and associated equipment, C: confidentiality, I: integrity, and A: availability.

References	Security Solution	Approach	Security Threats	Target Zone					Security Consideration		
				Z1	Z2	Z3	Z4	Z5	C	I	A
[129–131]	Drone security module	The drone security module encrypts the control signal and telemetry data from the UAV to the ground control station.	Unauthorized interception of encrypted data could compromise UAV operations.	✓	✓	✓	✓	X	✓	✓	X
[132–135]	Blockchain for secure data storage	Blockchain can be used to cryptographically store all the data that is sent to/from the drones.	Potential risks of data tampering despite the use of blockchain for storage.	✓	✓	✓	✓	✓	✓	✓	✓
[136–138]	Drone-assisted public safety networks	Unmanned aerial vehicles can be sent to suitable positions in the field to augment the operation of public safety networks.	Drones could be used maliciously to disrupt public safety networks.	✓	✓	✓	✓	X	X	✓	✓
[139–141]	Machine learning for threat detection	A machine learning solution based on a random forest classifier can be implemented to detect common network attacks such as denial of service, port scanning, and brute force.	Vulnerability to network attacks such as DoS, port scanning, and brute force.	✓	✓	✓	✓	✓	✓	✓	✓
[143–146]	Blockchain for UAV signal security	The use of blockchain technology when transmitting signals from the controller to the drone or UAV can achieve an extra amount of security when transmitting signals.	Despite blockchain usage, signal hijacking remains a critical concern.	✓	✓	✓	✓	X	✓	✓	X
[141,147,151]	Software-defined network for drone security (2023)	A software-defined network solution is suitable for a swarm of cooperative drones.	Coordinated attacks on drone swarms could lead to significant security breaches.	✓	✓	✓	✓	X	✓	✓	X
[148,149,152]	Light-weight hardware security	A light-weight hardware solution is proposed to assure the confidentiality and integrity of both the command data sent by the ground station and the payload data transmitted by the drone.	Hardware vulnerabilities could be exploited to compromise drone communications.	✓	✓	✓	✓	X	✓	✓	X
[150]	Multi-sensor detection systems	Utilizes multiple sensors to detect drones trespassing in protected areas and offers more compelling results.	Incomplete or failed detection of trespassing drones can pose serious security risks.	✓	X	X	✓	✓	X	✓	✓

The current landscape of drone security solutions leverages advanced cryptographic techniques, blockchain technology, machine learning, software-defined networking, and multi-sensor systems to address the diverse and evolving threats facing UAV operations. While these solutions collectively enhance the confidentiality, integrity, and availability of drone systems, ongoing research and development are essential to adapt to new security challenges and ensure robust protection for UAVs in various applications.

5. Future Research Directions

The integration of unmanned aerial vehicles (UAVs) into diverse sectors highlights the burgeoning need for robust cybersecurity measures and advanced forensic methodologies. This research has underscored the multifaceted nature of drone technology, where security concerns are as dynamic as the applications they support. As drones become integral to operations in agriculture, urban management, and public safety, the sophistication of cyber threats targeting these systems also escalates. The discussion of future research directions is thus critical to stay ahead of potential vulnerabilities that could compromise UAV operations and the data integrity they maintain.

Advancements in artificial intelligence (AI) and machine learning offer promising avenues for enhancing UAV security. Future research could focus on developing AI-driven anomaly detection systems that operate in real time to identify and mitigate threats before they can cause significant damage. These systems would benefit from continuous learning algorithms that adapt to new and evolving security threats, thereby maintaining the effectiveness of UAV operations across various environments.

Tamper-proof data chains for UAVs: such research would need to address scalability and efficiency and ensure that blockchain implementations do not impede the operational performance of drones, especially in time-sensitive applications like emergency response and real-time surveillance.

Standardization of forensic processes in the UAV domain remains a critical need. Research should aim to develop standardized protocols that not only facilitate effective forensic investigations but also ensure compliance with international laws and regulations. This includes creating guidelines for data collection, storage, and analysis that respect privacy concerns while providing clear and actionable insights during forensic investigations.

Furthermore, as the application of drones continues to expand, the development of predictive security measures to counteract future threats becomes more imperative. Research into predictive analytics frameworks that utilize vast amounts of operational data to forecast potential security breaches could significantly advance proactive security measures in UAV operations.

Collaboration across academic, industry, and regulatory domains is essential for advancing UAV security. Future research should encourage collaborative frameworks that allow for the sharing of knowledge, tools, and strategies. Such partnerships could accelerate the development of innovative solutions and help standardize security measures across borders and sectors.

In summary, enhancing UAV security and forensic capabilities requires a proactive and collaborative approach to research and development. By advancing AI and blockchain applications, standardizing forensic processes, and exploring predictive security measures, the UAV industry can safeguard against current and future cyber threats, ensuring the safe and efficient operation of drones in various sectors.

6. Conclusions

In conclusion, this study has provided a comprehensive examination of the cybersecurity vulnerabilities and forensic methodologies associated with unmanned aerial vehicles (UAVs) and underscores the critical need for robust security measures in an era of rapid technological advancement and widespread drone integration across various sectors. We have highlighted the importance of adopting a holistic approach to UAV security that combines advanced technological solutions with stringent regulatory frameworks to effectively mitigate potential threats.

We identified the integration of artificial intelligence and blockchain technology as particularly promising for enhancing the security and integrity of UAV operations. These technologies not only fortify defenses against cyber attacks but also streamline forensic processes, enabling quicker and more effective responses to incidents.

As UAVs continue to evolve and their applications expand, the landscape of potential cyber threats also broadens, necessitating a proactive approach to security. This involves not only defending against known vulnerabilities but also anticipating future challenges that could compromise UAV operations. Collaboration among academia, industry, and regulatory bodies is essential in this regard and fosters an environment of continuous improvement and adaptation to new security challenges.

The future of UAV security and forensics is dynamic and demands ongoing research and development. As this paper has shown, the strategic application of emerging technologies, combined with comprehensive policy-making and international cooperation, will be crucial in navigating the complexities of UAV security. Ensuring the safe and secure

operation of drones is not just about countering threats but also about harnessing their full potential to benefit society in diverse and meaningful ways.

Author Contributions: Conceptualization, A.A.; methodology, A.A.; validation, T.J.; formal analysis, A.A. and T.J.; writing—original draft preparation, A.A.; writing—review and editing, T.J.; visualization, A.A.; supervision, T.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new data were created or analyzed in this study.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Research Selection Criteria

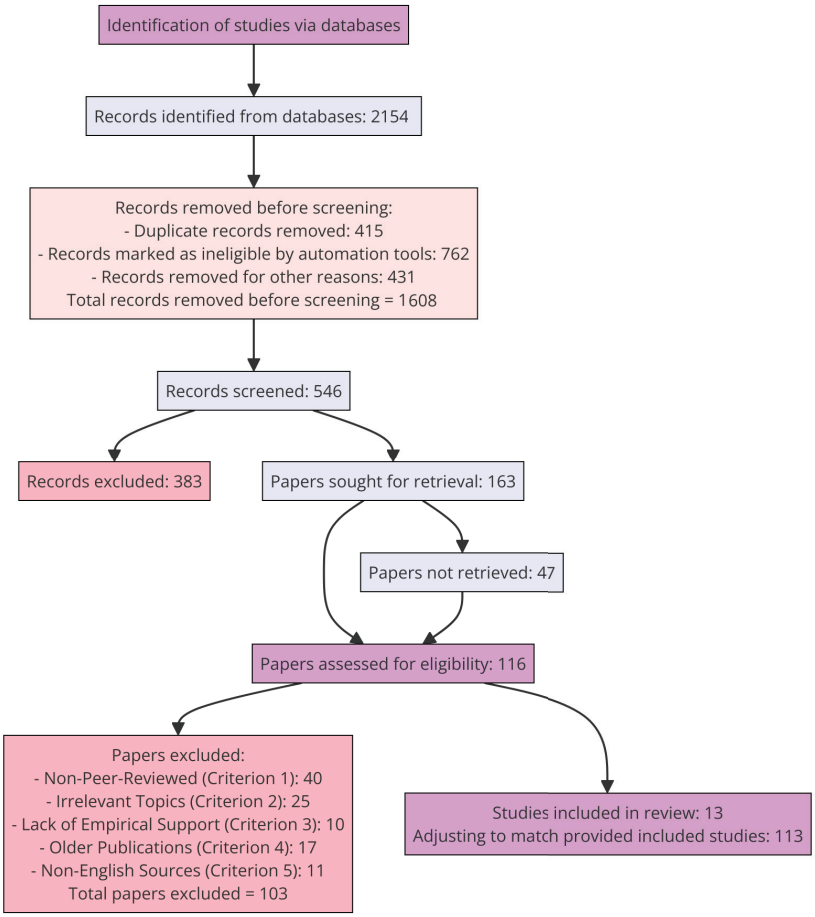


Figure A1. Search criteria process.

Appendix B. Research Inclusion and Exclusion Criteria

The inclusion and exclusion criteria are shared in Figures A2 and A3, respectively.

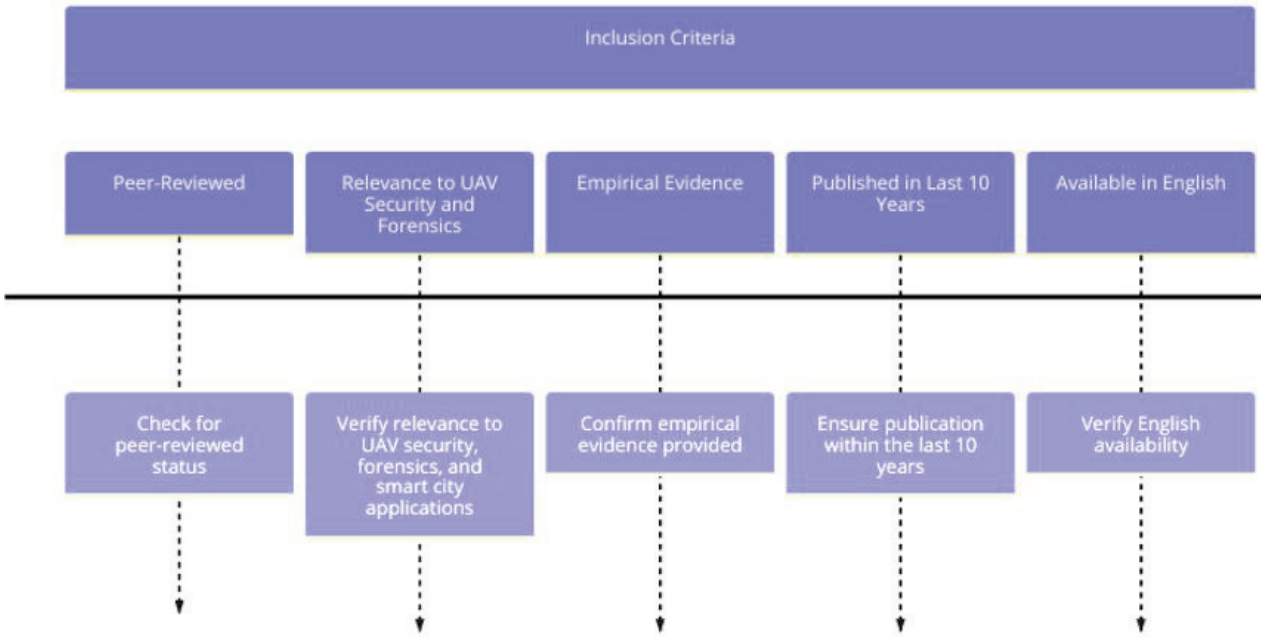


Figure A2. Inclusion criteria.

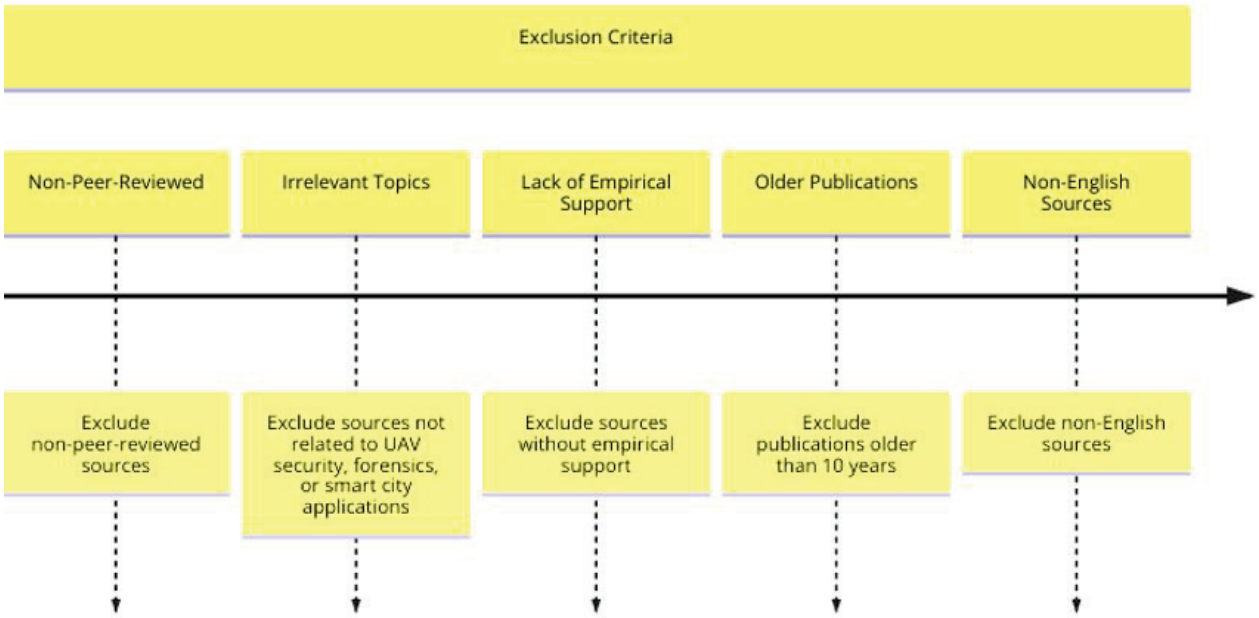


Figure A3. Exclusion criteria.

References

1. FAA Aerospace Forecast FY 2022-2042 | Federal Aviation Administration. Available online: https://www.faa.gov/dataresearch/aviation/faa-aerospace-forecast-fy-2022-2042 (accessed on 4 July 2023).
2. Sihag, V.; Choudhary, G.; Choudhary, P.; Dragoni, N. Cyber4Drone: A Systematic Review of Cyber Security and Forensics in Next-Generation Drones. *Drones* **2023**, *7*, 430. [CrossRef]
3. Ceviz, O.; Sadioglu, P.; Sen, S. A survey of security in uavs and fanets: Issues, threats, analysis of attacks, and solutions. *arXiv* **2023**, arXiv:2306.14281.
4. Shafik, W.; Matinkhah, S.M.; Shokoor, F. Cybersecurity in unmanned aerial vehicles: A review. *Int. J. Smart Sens. Intell. Syst.* **2023**, *16*, 12. [CrossRef]
5. Pandey, G.K.; Gurjar, D.S.; Nguyen, H.H.; Yadav, S. Security threats and mitigation techniques in UAV communications: A comprehensive survey. *IEEE Access* **2022**, *10*, 112858–112897. [CrossRef]

6. Siddiqi, M.A.; Iwendi, C.; Jaroslava, K.; Anumbe, N. Analysis on security-related concerns of unmanned aerial vehicle: Attacks, limitations, and recommendations. *Math. Biosci. Eng.* **2022**, *19*, 2641–2670. [CrossRef]
7. Wang, J.; Wang, X.; Gao, R.; Lei, C.; Feng, W.; Ge, N.; Jin, S.; Quek, T.Q. Physical layer security for UAV communications: A comprehensive survey. *China Commun.* **2022**, *19*, 77–115. [CrossRef]
8. Vergouw, B.; Nagel, H.; Bondt, G.; Custers, B. Drone technology: Types, payloads, applications, frequency spectrum issues and future developments. In *The Future of Drone Use: Opportunities and Threats from Ethical and Legal Perspectives*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–45. [CrossRef]
9. Shakhatreh, H.; Sawalmeh, A.H.; Al-Fuqaha, A.; Dou, Z.; Almaita, E.; Khalil, I.; Othman, N.S.; Khreishah, A.; Guizani, M. Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges. *IEEE Access* **2019**, *7*, 48572–48634. [CrossRef]
10. Amici, C.; Ceresoli, F.; Pasetti, M.; Saponi, M.; Tiboni, M.; Zanoni, S. Review of propulsion system design strategies for unmanned aerial vehicles. *Appl. Sci.* **2021**, *11*, 5209. [CrossRef]
11. Chiper, F.L.; Martian, A.; Vladeanu, C.; Marghescu, I.; Craciunescu, R.; Fratu, O. Drone detection and defense systems: Survey and a software-defined radio-based solution. *Sensors* **2022**, *22*, 1453. [CrossRef]
12. Joshi, D.; Deb, D.; Muyeen, S. Comprehensive review on electric propulsion system of unmanned aerial vehicles. *Front. Energy Res.* **2022**, *10*, 752012. [CrossRef]
13. Velusamy, P.; Rajendran, S.; Mahendran, R.K.; Naseer, S.; Shafiq, M.; Choi, J.G. Unmanned Aerial Vehicles (UAV) in precision agriculture: Applications and challenges. *Energies* **2021**, *15*, 217. [CrossRef]
14. Mohsan, S.A.H.; Othman, N.Q.H.; Li, Y.; Alsharif, M.H.; Khan, M.A. Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends. *Intell. Serv. Robot.* **2023**, *16*, 109–137. [CrossRef] [PubMed]
15. Fraser, B.T.; Congalton, R.G. Issues in Unmanned Aerial Systems (UAS) data collection of complex forest environments. *Remote Sens.* **2018**, *10*, 908. [CrossRef]
16. Apostolidis, S.D.; Kapoutsis, P.C.; Kapoutsis, A.C.; Kosmatopoulos, E.B. Cooperative multi-UAV coverage mission planning platform for remote sensing applications. *Auton. Robot.* **2022**, *46*, 373–400. [CrossRef]
17. Rakha, T.; Gorodetsky, A. Review of Unmanned Aerial System (UAS) applications in the built environment: Towards automated building inspection procedures using drones. *Autom. Constr.* **2018**, *93*, 252–264. [CrossRef]
18. Sharma, A.; Vanjani, P.; Paliwal, N.; Basnayaka, C.M.W.; Jayakody, D.N.K.; Wang, H.C.; Muthuchidambaranathan, P. Communication and networking technologies for UAVs: A survey. *J. Netw. Comput. Appl.* **2020**, *168*, 102739. [CrossRef]
19. Areias, B.; Martins, A.; Paula, N.; Reis, A.B.; Sargento, S. A control and communications platform for procedural mission planning with multiple aerial drones. *Pers. Ubiquitous Comput.* **2022**, *26*, 1105–1115. [CrossRef]
20. Viswanathan, S.; Baig, Z. Digital forensics for drones: A study of tools and techniques. In Proceedings of the Applications and Techniques in Information Security: 11th International Conference, ATIS 2020, Brisbane, QLD, Australia, 12–13 November 2020; Proceedings 11; Springer: Berlin/Heidelberg, Germany, 2020; pp. 29–41. [CrossRef]
21. Barton, T.E.A.; Azhar, M.H.B. Forensic analysis of popular UAV systems. In Proceedings of the 7th International Conference on Emerging Security Technologies (EST), Canterbury, UK, 6–8 September 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 91–96. [CrossRef]
22. Barton, T.E.A.; Azhar, M.H.B. Open source forensics for a multi-platform drone system. In Proceedings of the Digital Forensics and Cyber Crime: 9th International Conference, ICDF2C 2017, Prague, Czech Republic, 9–11 October 2017; Proceedings 9; Springer: Berlin/Heidelberg, Germany, 2018; pp. 83–96. [CrossRef]
23. Roder, A.; Choo, K.K.R.; Le-Khac, N.A. Unmanned aerial vehicle forensic investigation process: Dji phantom 3 drone as a case study. *arXiv* **2018**, arXiv:1804.08649.
24. Mekala, S.H.; Baig, Z. Digital forensics for drone data–intelligent clustering using self organising maps. In Proceedings of the Future Network Systems and Security: 5th International Conference, FNSS 2019, Melbourne, VIC, Australia, 27–29 November 2019; Proceedings 5; Springer: Berlin/Heidelberg, Germany, 2019; pp. 172–189. [CrossRef]
25. Siddappaji, B.; Akhilesh, K. Role of cyber security in drone technology. In *Smart Technologies: Scope and Applications*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 169–178. [CrossRef]
26. Nguyen, H.P.D.; Nguyen, D.D. Drone application in smart cities: The general overview of security vulnerabilities and countermeasures for data communication. In *Development and Future of Internet of Drones (IoD): Insights, Trends and Road Ahead*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 185–210. [CrossRef]
27. Yahuza, M.; Idris, M.Y.I.; Ahmedy, I.B.; Wahab, A.W.A.; Nandy, T.; Noor, N.M.; Bala, A. Internet of drones security and privacy issues: Taxonomy and open challenges. *IEEE Access* **2021**, *9*, 57243–57270. [CrossRef]
28. Alhussan, A.A.; Al-Dhaqm, A.; Yafooz, W.M.; Razak, S.B.A.; Emara, A.H.M.; Khafaga, D.S. Towards development of a high abstract model for drone forensic domain. *Electronics* **2022**, *11*, 1168. [CrossRef]
29. Bouafif, H.; Kamoun, F.; Iqbal, F. Towards a better understanding of drone forensics: A case study of parrot AR drone 2.0. *Int. J. Digit. Crime Forensics* **2020**, *12*, 35–57. [CrossRef]
30. Yaacoub, J.P.; Noura, H.; Salman, O.; Chehab, A. Security analysis of drones systems: Attacks, limitations, and recommendations. *Internet Things* **2020**, *11*, 100218. [CrossRef]
31. Khan, S.Z.; Mohsin, M.; Iqbal, W. On GPS spoofing of aerial platforms: A review of threats, challenges, methodologies, and future research directions. *PeerJ Comput. Sci.* **2021**, *7*, e507. [CrossRef]

32. Davidovich, B.; Nassi, B.; Elovici, Y. Towards the detection of GPS spoofing attacks against drones by analyzing camera's video stream. *Sensors* **2022**, *22*, 2608. [CrossRef] [PubMed]
33. Altaweel, A.; Mukkath, H.; Kamel, I. GPS Spoofing attacks in FANETs: A systematic literature review. *IEEE Access* **2023**, *11*, 55233–55280. [CrossRef]
34. Pardhasaradhi, B.; Cenkeramaddi, L.R. GPS spoofing detection and mitigation for drones using distributed radar tracking and fusion. *IEEE Sens. J.* **2022**, *22*, 11122–11134. [CrossRef]
35. Greco, C.; Pace, P.; Basagni, S.; Fortino, G. Jamming detection at the edge of drone networks using multi-layer perceptrons and decision trees. *Appl. Soft Comput.* **2021**, *111*, 107806. [CrossRef]
36. Arthur, M.P. Detecting signal spoofing and jamming attacks in UAV networks using a lightweight IDS. In Proceedings of the 2019 International Conference on Computer, Information and Telecommunication Systems (CITS), Beijing, China, 28–31 August 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–5. [CrossRef]
37. Di Pietro, R.; Oligeri, G.; Tedeschi, P. Jam-me: Exploiting jamming to accomplish drone mission. In Proceedings of the 2019 IEEE Conference on Communications and Network Security (CNS), Washington, DC, USA, 10–12 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–9. [CrossRef]
38. Mekdad, Y.; Acar, A.; Aris, A.; Fergougui, A.E.; Conti, M.; Lazzeretti, R.; Uluagac, S. Exploring Jamming and Hijacking Attacks for Micro Aerial Drones. *arXiv* **2024**, arXiv:2403.03858.
39. Pawlak, J.; Li, Y.; Price, J.; Wright, M.; Al Shamaileh, K.; Niyaz, Q.; Devabhaktuni, V. A machine learning approach for detecting and classifying jamming attacks against ofdm-based uavs. In Proceedings of the 3rd ACM Workshop on Wireless Security and Machine Learning, Abu Dhabi, United Arab Emirates, 28 June–2 July 2021; pp. 1–6. [CrossRef]
40. Hartmann, K.; Giles, K. UAV exploitation: A new domain for cyber power. In Proceedings of the 8th International Conference on Cyber Conflict (CyCon), Tallinn, Estonia, 31 May–3 June 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 205–221. [CrossRef]
41. Kong, P.Y. A survey of cyberattack countermeasures for unmanned aerial vehicles. *IEEE Access* **2021**, *9*, 148244–148263. [CrossRef]
42. Lykou, G.; Moustakas, D.; Gritzalis, D. Defending airports from UAS: A survey on cyber-attacks and counter-drone sensing technologies. *Sensors* **2020**, *20*, 3537. [CrossRef]
43. Mekdad, Y.; Aris, A.; Babun, L.; El Fergougui, A.; Conti, M.; Lazzeretti, R.; Uluagac, A.S. A survey on security and privacy issues of UAVs. *Comput. Netw.* **2023**, *224*, 109626. [CrossRef]
44. Ly, B.; Ly, R. Cybersecurity in unmanned aerial vehicles (UAVs). *J. Cyber Secur. Technol.* **2021**, *5*, 120–137. [CrossRef]
45. Baig, Z.; Syed, N.; Mohammad, N. Securing the smart city airspace: Drone cyber attack detection through machine learning. *Future Internet* **2022**, *14*, 205. [CrossRef]
46. Liao, L.; Xie, F.; Chen, J. Analysis on Technology of High-Energy Counter-UAVs Laser Weapon. In Proceedings of the 4th International Conference on Vision, Image and Signal Processing, Bangkok, Thailand, 9–11 December 2020; pp. 1–5. [CrossRef]
47. Lyu, C.; Zhan, R. Global analysis of active defense technologies for unmanned aerial vehicle. *IEEE Aerosp. Electron. Syst. Mag.* **2022**, *37*, 6–31. [CrossRef]
48. Ahmed, S.A.; Mohsin, M.; Ali, S.M.Z. Survey and technological analysis of laser and its defense applications. *Def. Technol.* **2021**, *17*, 583–592. [CrossRef]
49. Chaari, M.Z.; Al-Maadeed, S. The game of drones/weapons makers' war on drones. In *Unmanned Aerial Systems*; Elsevier: Amsterdam, The Netherlands, 2021; pp. 465–493. [CrossRef]
50. Kaushal, H.; Kaddoum, G. Applications of lasers for tactical military operations. *IEEE Access* **2017**, *5*, 20736–20753. [CrossRef]
51. Omolara, A.E.; Alawida, M.; Abiodun, O.I. Drone cybersecurity issues, solutions, trend insights and future perspectives: A survey. *Neural Comput. Appl.* **2023**, *35*, 23063–23101. [CrossRef]
52. Manesh, M.R.; Kaabouch, N. Cyber-attacks on unmanned aerial system networks: Detection, countermeasure, and future research directions. *Comput. Secur.* **2019**, *85*, 386–401. [CrossRef]
53. Vattapparamban, E.; Güvenç, I.; Yurekli, A.I.; Akkaya, K.; Uluağaç, S. Drones for smart cities: Issues in cybersecurity, privacy, and public safety. In Proceedings of the 2016 International Wireless Communications and Mobile Computing Conference (IWCMC), Paphos, Cyprus, 5–9 September 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 216–221. [CrossRef]
54. Rugo, A.; Ardagna, C.A.; Ioini, N.E. A security review in the UAVNet era: Threats, countermeasures, and gap analysis. *ACM Comput. Surv.* **2022**, *55*, 1–35. [CrossRef]
55. Vajravelu, A.; Ashok Kumar, N.; Sarkar, S.; Degadwala, S. Security threats of unmanned aerial vehicles. In *Wireless Networks: Cyber Security Threats and Countermeasures*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 133–164. [CrossRef]
56. Wang, J.; Liu, Y.; Song, H. Counter-unmanned aircraft system (s)(C-UAS): State of the art, challenges, and future trends. *IEEE Aerosp. Electron. Syst. Mag.* **2021**, *36*, 4–29. [CrossRef]
57. Gabrielsson, J.; Bugeja, J.; Vogel, B. Hacking a Commercial drone with open-source software: Exploring data privacy violations. In Proceedings of the 10th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 7–10 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–5. [CrossRef]
58. Rubbestad, G.; Söderqvist, W. Hacking a Wi-Fi Based Drone. 2021. Available online: https://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-299887 (accessed on 17 May 2024).
59. Gabrielsson, J. Hacking Your Drone Data. 2021. Available online: https://urn.kb.se/resolve?urn=urn:nbn:se:mau:diva-41403 (accessed on 17 May 2024).

60. Westerlund, O.; Asif, R. Drone hacking with raspberry-pi 3 and wifi pineapple: Security and privacy threats for the internet-of-things. In Proceedings of the 1st International Conference on Unmanned Vehicle Systems-Oman (UVS), Muscat, Oman, 5–7 February 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–10. [CrossRef]
61. Karmakar, G.; Petty, M.; Ahmed, H.; Das, R.; Kamruzzaman, J. Security of Internet of Things Devices: Ethical Hacking a Drone and its Mitigation Strategies. In Proceedings of the 2022 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Gold Coast, Australia, 18–20 December 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–5. [CrossRef]
62. Astaburuaga, I.; Lombardi, A.; La Torre, B.; Hughes, C.; Sengupta, S. Vulnerability analysis of ar. drone 2.0, an embedded linux system. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 666–672. [CrossRef]
63. Lakew Yihunie, F.; Singh, A.K.; Bhatia, S. Assessing and exploiting security vulnerabilities of unmanned aerial vehicles. In *Proceedings of the Smart Systems and IoT: Innovations in Computing: Proceeding of SSIC 2019*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 701–710. [CrossRef]
64. El-Rewini, Z.; Sadatsharan, K.; Sugunaraj, N.; Selvaraj, D.F.; Plathottam, S.J.; Ranganathan, P. Cybersecurity attacks in vehicular sensors. *IEEE Sens. J.* **2020**, *20*, 13752–13767. [CrossRef]
65. Teixidó, P.; Gómez-Galán, J.A.; Caballero, R.; Pérez-Grau, F.J.; Hinojo-Montero, J.M.; Muñoz-Chavero, F.; Aponte, J. Secured perimeter with electromagnetic detection and tracking with drone embedded and static cameras. *Sensors* **2021**, *21*, 7379. [CrossRef] [PubMed]
66. OConnor, T.; Enck, W.; Reaves, B. Blinded and confused: Uncovering systemic flaws in device telemetry for smart-home internet of things. In Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, Miami, FL, USA, 15–17 May 2019; pp. 140–150. [CrossRef]
67. Rong-Xiao, G.; Ji-wei, T.; Bu-hong, W.; Fu-te, S. Cyber-physical attack threats analysis for UAVs from CPS perspective. In Proceedings of the 2020 International Conference on Computer Engineering and Application (ICCEA), Guangzhou, China, 18–20 March 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 259–263. [CrossRef]
68. Wang, Z.; Li, Y.; Wu, S.; Zhou, Y.; Yang, L.; Xu, Y.; Zhang, T.; Pan, Q. A survey on cybersecurity attacks and defenses for unmanned aerial systems. *J. Syst. Archit.* **2023**, *138*, 102870. [CrossRef]
69. Salamh, F.E.; Karabiyik, U.; Rogers, M.; Al-Hazemi, F. Drone disrupted denial of service attack (3DOS): Towards an incident response and forensic analysis of remotely piloted aerial systems (RPASs). In Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 24–28 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 704–710. [CrossRef]
70. Ouiazzane, S.; Addou, M.; Barramou, F. A multiagent and machine learning based denial of service intrusion detection system for drone networks. In *Geospatial Intelligence: Applications and Future Trends*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 51–65. [CrossRef]
71. De Carvalho Bertoli, G.; Pereira, L.A.; Saotome, O. Classification of denial of service attacks on Wi-Fi-based unmanned aerial vehicle. In Proceedings of the 2021 10th Latin-American Symposium on Dependable Computing (LADC), Florianópolis, Brazil, 22–26 November 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6. [CrossRef]
72. Vasconcelos, G.; Carrijo, G.; Miani, R.; Souza, J.; Guizilini, V. The impact of DoS attacks on the AR. Drone 2.0. In Proceedings of the 2016 13th Latin American Robotics Symposium and 4th Brazilian Robotics Symposium (LARS/SBR), Recife, Brazil, 8–12 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 127–132. [CrossRef]
73. Krichen, M.; Adoni, W.Y.H.; Mihoub, A.; Alzahrani, M.Y.; Nahhal, T. Security challenges for drone communications: Possible threats, attacks and countermeasures. In Proceedings of the 2022 2nd International Conference of Smart Systems and Emerging Technologies (SMARTTECH), Riyadh, Saudi Arabia, 9–11 May 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 184–189. [CrossRef]
74. Elkhider, S.M.; El-Ferik, S.; Saif, A.W.A. Containment control of multiagent systems subject to denial of service attacks. *IEEE Access* **2022**, *10*, 48102–48111. [CrossRef]
75. Orhun, D.; Karakoca, Y.E.; Camadan, E.; Baykali, F. Hybrid cyber security of unmanned aerial vehicles. *Int. J. Appl. Methods Electron. Comput.* **2023**, *11*, 179–185. [CrossRef]
76. Abdulrazak, C. Cybersecurity Threat Analysis And Attack Simulations For Unmanned Aerial Vehicle Networks. *arXiv* **2024**, arXiv:2404.16842.
77. Dwivedi, K.; Govindarajan, P.; Srinivasan, D.; Keerthi Sanjana, A.; Selvanambi, R.; Karuppiah, M. Intelligent autonomous drones in Industry 4.0. In *Artificial Intelligence and Cyber Security in Industry 4.0*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 133–163. [CrossRef]
78. Schmitt, C.; Körner, J.; Leuck, S. PSAT–A Package Structure Analyzation Tool to Regain Control of Hijacked Drones. In Proceedings of the 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC), Barcelona, Spain, 1–5 October 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–10. [CrossRef]
79. Mohammed, A.B.; Fourati, L.C.; Fakhrudeen, A.M. Comprehensive systematic review of intelligent approaches in UAV-based intrusion detection, blockchain, and network security. *Comput. Netw.* **2023**, *239*, 110140. [CrossRef]
80. Kolisnyk, M.; Piskachov, O. Analysis and Systematization of Vulnerabilities of Drone Subsystems. In *Proceedings of the International Conference on Information and Communication Technologies in Education, Research, and Industrial Applications*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 65–81. [CrossRef]

81. Chaudhary, D.; Soni, T.; Vasudev, K.L.; Saleem, K. A modified lightweight authenticated key agreement protocol for Internet of Drones. *Internet Things* **2023**, *21*, 100669. [CrossRef]
82. Mahmood, K.; Ghaffar, Z.; Farooq, M.; Yahya, K.; Das, A.K.; Chaudhry, S.A. A Security Enhanced Chaotic-Map Based Authentication Protocol for Internet of Drones. *IEEE Internet Things J.* **2024**, *11*, 22301–22309. [CrossRef]
83. Jamil, A.M.; Hadi, H.J.; Li, S.; Cao, Y.; Ahmed, N.; Hussain, F.B.; Suthaputchakun, C.; Wang, X. Detection of Targeted Attacks Using Medium-Interaction Honeypot for Unmanned Aerial Vehicle. In *Proceedings of the International Conference on Digital Forensics and Cyber Crime*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 164–185. [CrossRef]
84. Costin, A.; Khandker, S.; Turtiainen, H.; Hämäläinen, T. Cybersecurity of COSPAS-SARSAT and EPIRB: Threat and attacker models, exploits, future research. *arXiv* **2023**, arXiv:2302.08361.
85. Yasmine, G.; Maha, G.; Alaoui, A.E.H. Anti-drone systems: Current intelligent countermeasures from low to high risks. In Proceedings of the 2023 7th IEEE Congress on Information Science and Technology (CiSt), Agadir-Essaouira, Morocco, 16–22 December 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 317–322. [CrossRef]
86. Ardizzon, F.; Salvaterra, D.; Piana, M.; Tomasin, S. Energy-Based Optimization of Physical-Layer Challenge-Response Authentication with Drones. *arXiv* **2024**, arXiv:2405.03608.
87. Li, Z.; Chen, Q.; Mo, W.; Wang, X.; Hu, L.; Cao, Y. Converging Blockchain and Deep Learning in UAV Network Defense Strategy: Ensuring Data Security During Flight. In *Proceedings of the International Conference on Artificial Intelligence Security and Privacy*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 156–171. [CrossRef]
88. Sharma, J.; Mehra, P.S. Secure communication in IOT-based UAV networks: A systematic survey. *Internet Things* **2023**, *23*, 100883. [CrossRef]
89. Sharma, S.; Sarangi, P.K.; Sharma, B.; Subudhi, G.B. Implementation Analysis of Ransomware and Unmanned Aerial Vehicle Attacks: Mitigation Methods and UAV Security Recommendations. In *Advances in Aerial Sensing and Imaging*; Wiley and Sons: Hoboken, NJ, USA, 2024; pp. 165–211. [CrossRef]
90. Bouke, M.A.; Abdullah, A. SMRD: A Novel Cyber Warfare Modeling Framework for Social Engineering, Malware, Ransomware, and Distributed Denial-of-Service Based on a System of Nonlinear Differential Equations. *J. Appl. Artif. Intell.* **2024**, *5*, 54–68. [CrossRef]
91. Almerza, N. Agent-Based Modeling to Determine the Risk to a Swarm of Unmanned Aerial Vehicles under an Adversarial Artificial Intelligence Attack. Ph.D. Thesis, Marymount University, Arlington, VA, USA, 2023. Available online: https://www.proquest.com/docview/2828097307/abstract/477A601D16F34C9DPQ/1 (accessed on 17 May 2024).
92. Hoang, T.M.; Nguyen, N.M.; Duong, T.Q. Detection of eavesdropping attack in UAV-aided wireless systems: Unsupervised learning with one-class SVM and k-means clustering. *IEEE Wirel. Commun. Lett.* **2019**, *9*, 139–142. [CrossRef]
93. Shaikhanov, Z.; Badran, S.; Jornet, J.M.; Mittleman, D.M.; Knightly, E.W. Remotely positioned metasurface-drone attack. In Proceedings of the 24th International Workshop on Mobile Computing Systems and Applications, Newport Beach, CA, USA, 22–23 February 2023; pp. 110–116. [CrossRef]
94. Hassija, V.; Chamola, V.; Agrawal, A.; Goyal, A.; Luong, N.C.; Niyato, D.; Yu, F.R.; Guizani, M. Fast, reliable, and secure drone communication: A comprehensive survey. *IEEE Commun. Surv. Tutorials* **2021**, *23*, 2802–2832. [CrossRef]
95. Li, K.; Voicu, R.C.; Kanhere, S.S.; Ni, W.; Tovar, E. Energy efficient legitimate wireless surveillance of UAV communications. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2283–2293. [CrossRef]
96. AL-Dosari, K.; Deif, A.M.; Kucukvar, M.; Onat, N.; Fetais, N. Security Supply Chain Using UAVs: Validation and Development of a UAV-Based Model for Qatar's Mega Sporting Events. *Drones* **2023**, *7*, 555. [CrossRef]
97. Sobb, T.; Turnbull, B.; Moustafa, N. Supply chain 4.0: A survey of cyber security challenges, solutions and future directions. *Electronics* **2020**, *9*, 1864. [CrossRef]
98. Sontowski, S.; Gupta, M.; Chukkapalli, S.S.L.; Abdelsalam, M.; Mittal, S.; Joshi, A.; Sandhu, R. Cyber attacks on smart farming infrastructure. In Proceedings of the 2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC), Atlanta, GA, USA, 1–3 December 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 135–143. [CrossRef]
99. Pyzynski, M.; Balcerzak, T. Cybersecurity of the unmanned aircraft system (UAS). *J. Intell. Robot. Syst.* **2021**, *102*, 35. [CrossRef]
100. Vacek, J.J. The next frontier in drone law: Liability for cybersecurity negligence and data breaches for UAS operators. *Campbell Law Rev.* **2017**, *39*, 135.
101. Jacobsen, R.H.; Marandi, A. Security threats analysis of the unmanned aerial vehicle system. In Proceedings of the MILCOM 2021–2021 IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 29 November–2 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 316–322. [CrossRef]
102. Kulp, P.; Mei, N. A framework for sensing radio frequency spectrum attacks on medical delivery drones. In Proceedings of the 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 11–14 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 408–413. [CrossRef]
103. Rural, S. *Current and Future Regulatory Requirements that Impact on the Safe Commercial and Recreational Use of Remotely Piloted Aircraft Systems (RPAS), Unmanned Aerial Systems (UAS) and Associated Systems*; Parliament of Australia: Canberra, ACT, Australia, 2018.
104. Swan, R. *Drone Strikes: An Overview, Articulation and Assessment of the United States' Position under International Law*; Lawrence Livermore National Laboratory: Livermore, CA, USA, 2019. [CrossRef]
105. Kwik, J. Mitigating the Risk of Autonomous Weapon Misuse by Insurgent Groups. *Laws* **2023**, *12*, 5. [CrossRef]
106. Wesson, K.; Humphreys, T. Hacking drones. *Sci. Am.* **2013**, *309*, 54–59. [CrossRef] [PubMed]

107. Bunse, C.; Plotz, S. Security analysis of drone communication protocols. In Proceedings of the Engineering Secure Software and Systems: 10th International Symposium, ESSoS 2018, Paris, France, 26–27 June 2018; Proceedings 10; Springer: Berlin/Heidelberg, Germany, 2018; pp. 96–107. [CrossRef]
108. Tedeschi, P.; Oligeri, G.; Di Pietro, R. Leveraging jamming to help drones complete their mission. *IEEE Access* **2019**, *8*, 5049–5064. [CrossRef]
109. Kou, L.; Ding, S.; Wu, T.; Dong, W.; Yin, Y. An intrusion detection model for drone communication network in SDN environment. *Drones* **2022**, *6*, 342. [CrossRef]
110. Alhawi, O.M.; Mustafa, M.A.; Cordiro, L.C. Finding security vulnerabilities in unmanned aerial vehicles using software verification. In Proceedings of the 2019 International Workshop on Secure Internet of Things (SIOT), Luxembourg, 26 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–9. [CrossRef]
111. Oláh, N.; Molnár, B.; Huszti, A. Secure Registration Protocol for the Internet of Drones Using Blockchain and Physical Unclonable Function Technology. *Symmetry* **2023**, *15*, 1886. [CrossRef]
112. Guvenc, I.; Koohifar, F.; Singh, S.; Sichitiu, M.L.; Matolak, D. Detection, tracking, and interdiction for amateur drones. *IEEE Commun. Mag.* **2018**, *56*, 75–81. [CrossRef]
113. Rejeb, A.; Rejeb, K.; Simske, S.J.; Treiblmaier, H. Drones for supply chain management and logistics: A review and research agenda. *Int. J. Logist. Res. Appl.* **2023**, *26*, 708–731. [CrossRef]
114. Renduchintala, A.L.S.; Albehadili, A.; Javaid, A.Y. Drone forensics: Digital flight log examination framework for micro drones. In Proceedings of the 2017 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 14–16 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 91–96. [CrossRef]
115. Azhar, M.; Barton, T.E.A.; Islam, T. Drone forensic analysis using open source tools. *J. Digit. Forensics Secur. Law* **2018**, *13*, 6. [CrossRef]
116. Bouafif, H.; Kamoun, F.; Iqbal, F.; Marrington, A. Drone forensics: Challenges and new insights. In Proceedings of the 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, France, 26–28 February 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–6. [CrossRef]
117. Iqbal, F.; Yankson, B.; AlYammahi, M.A.; AlMansoori, N.; Qayed, S.M.; Shah, B.; Baker, T. Drone forensics: Examination and analysis. *Int. J. Electron. Secur. Digit. Forensics* **2019**, *11*, 245–264. [CrossRef]
118. Kao, D.Y.; Chen, M.C.; Wu, W.Y.; Lin, J.S.; Chen, C.H.; Tsai, F. Drone forensic investigation: DJI spark drone as a case study. *Procedia Comput. Sci.* **2019**, *159*, 1890–1899. [CrossRef]
119. Mantas, E.; Patsakis, C. GRYPHON: Drone forensics in dataflash and telemetry logs. In Proceedings of the Advances in Information and Computer Security: 14th International Workshop on Security, IWSEC 2019, Tokyo, Japan, 28–30 August 2019; Proceedings 14; Springer: Berlin/Heidelberg, Germany, 2019; pp. 377–390. [CrossRef]
120. Yousef, M.; Iqbal, F.; Hussain, M. Drone forensics: A detailed analysis of emerging DJI models. In Proceedings of the 2020 11th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 7–9 April 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 66–71. [CrossRef]
121. Al-Room, K.; Iqbal, F.; Baker, T.; Shah, B.; Yankson, B.; MacDermott, A.; Hung, P.C. Drone forensics: A case study of digital forensic investigations conducted on common drone models. *Int. J. Digit. Crime Forensics* **2021**, *13*, 1–25. [CrossRef]
122. Al-Dhaqm, A.; Ikuesan, R.A.; Kebande, V.R.; Razak, S.; Ghabban, F.M. Research challenges and opportunities in drone forensics models. *Electronics* **2021**, *10*, 1519. [CrossRef]
123. Alotaibi, F.M.; Al-Dhaqm, A.; Al-Otaibi, Y.D. A novel forensic readiness framework applicable to the drone forensics field. *Comput. Intell. Neurosci.* **2022**, *2022*, e8002963. [CrossRef]
124. Lan, J.K.W.; Lee, F.K.W. Drone forensics: A case study on dji mavic air 2. In Proceedings of the 2022 24th International Conference on Advanced Communication Technology (ICACT), PyeongChang, Republic of Korea, 13–16 February 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 291–296. [CrossRef]
125. Da Silva, L.M.; Menezes, H.B.d.B.; Luccas, M.d.S.; Mailer, C.; Pinto, A.S.R.; Boava, A.; Rodrigues, M.; Ferrão, I.G.; Estrella, J.C.; Branco, K.R.L.J.C. Development of an efficiency platform based on MQTT for UAV controlling and DoS attack detection. *Sensors* **2022**, *22*, 6567. [CrossRef] [PubMed]
126. Buccafurri, F.; De Angelis, V.; Lazzaro, S. MQTT-A: A broker-bridging P2P architecture to achieve anonymity in MQTT. *IEEE Internet Things J.* **2023**, *10*, 15443–15463. [CrossRef]
127. Xiong, F.; Li, A.; Wang, H.; Tang, L. An SDN-MQTT based communication system for battlefield UAV swarms. *IEEE Commun. Mag.* **2019**, *57*, 41–47. [CrossRef]
128. Baig, Z.; Khan, M.A.; Mohammad, N.; Brahim, G.B. Drone forensics and machine learning: Sustaining the investigation process. *Sustainability* **2022**, *14*, 4861. [CrossRef]
129. Schiller, N.; Chlosta, M.; Schloegel, M.; Bars, N.; Eisenhofer, T.; Scharnowski, T.; Domke, F.; Schönherr, L.; Holz, T. Drone Security and the Mysterious Case of DJI's DroneID. In Proceedings of the Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 27 February–3 March 2023.
130. Kim, K.; Kang, Y. Drone security module for UAV data encryption. In Proceedings of the 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Republic of Korea, 21–23 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1672–1674. [CrossRef]

131. Samanth, S.; Prema, K.V.; Balachandra, M. Security in internet of drones: A comprehensive review. *Cogent Eng.* **2022**, *9*, 2029080. [CrossRef]
132. Cheema, M.A.; Ansari, R.I.; Ashraf, N.; Hassan, S.A.; Qureshi, H.K.; Bashir, A.K.; Politis, C. Blockchain-based secure delivery of medical supplies using drones. *Comput. Netw.* **2022**, *204*, 108706. [CrossRef]
133. Bera, B.; Chattaraj, D.; Das, A.K. Designing secure blockchain-based access control scheme in IoT-enabled Internet of Drones deployment. *Comput. Commun.* **2020**, *153*, 229–249. [CrossRef]
134. Singh, M.; Aujla, G.S.; Bali, R.S. A deep learning-based blockchain mechanism for secure internet of drones environment. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 4404–4413. [CrossRef]
135. Gupta, R.; Kumari, A.; Tanwar, S. Fusion of blockchain and artificial intelligence for secure drone networking underlying 5G communications. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e4176. [CrossRef]
136. Minhas, H.I.; Ahmad, R.; Ahmed, W.; Waheed, M.; Alam, M.M.; Gul, S.T. A reinforcement learning routing protocol for UAV aided public safety networks. *Sensors* **2021**, *21*, 4121. [CrossRef] [PubMed]
137. He, D.; Chan, S.; Guizani, M. Drone-assisted public safety networks: The security aspect. *IEEE Commun. Mag.* **2017**, *55*, 218–223. [CrossRef]
138. Ali, K.; Nguyen, H.X.; Vien, Q.T.; Shah, P.; Raza, M.; Paranthaman, V.V.; Er-Rahmadi, B.; Awais, M.; ul Islam, S.; Rodrigues, J.J. Review and implementation of resilient public safety networks: 5G, IoT, and emerging technologies. *IEEE Netw.* **2021**, *35*, 18–25. [CrossRef]
139. Studiawan, H.; Grispos, G.; Choo, K.K.R. Unmanned Aerial Vehicle (UAV) Forensics: The Good, The Bad, and the Unaddressed. *Comput. Secur.* **2023**, *132*, 103340. [CrossRef]
140. Abu Al-Haija, Q.; Al Badawi, A. High-performance intrusion detection system for networked UAVs via deep learning. *Neural Comput. Appl.* **2022**, *34*, 10885–10900. [CrossRef]
141. Guerber, C.; Royer, M.; Larrieu, N. Machine Learning and Software Defined Network to secure communications in a swarm of drones. *J. Inf. Secur. Appl.* **2021**, *61*, 102940. [CrossRef]
142. Heidari, A.; Jafari Navimipour, N.; Unal, M.; Zhang, G. Machine learning applications in internet-of-drones: Systematic review, recent deployments, and open issues. *ACM Comput. Surv.* **2023**, *55*, 1–45. [CrossRef]
143. Hafeez, S.; Khan, A.R.; Al-Quraan, M.; Mohjazi, L.; Zoha, A.; Imran, M.A.; Sun, Y. Blockchain-assisted UAV communication systems: A comprehensive survey. *IEEE Open J. Veh. Technol.* **2023**, *4*, 558–580. [CrossRef]
144. Kumar, R.; Aljuhani, A.; Kumar, P.; Kumar, A.; Franklin, A.; Jolfaei, A. Blockchain-enabled secure communication for unmanned aerial vehicle (UAV) networks. In Proceedings of the 5th International ACM Mobicom Workshop on Drone Assisted Wireless Communications for 5G and Beyond, Sydney, NSW, Australia, 17 October 2022; pp. 37–42.
145. Ch, R.; Srivastava, G.; Gadekallu, T.R.; Maddikunta, P.K.R.; Bhattacharya, S. Security and privacy of UAV data using blockchain technology. *J. Inf. Secur. Appl.* **2020**, *55*, 102670. [CrossRef]
146. Rana, T.; Shankar, A.; Sultan, M.K.; Patan, R.; Balusamy, B. An intelligent approach for UAV and drone privacy security using blockchain methodology. In Proceedings of the 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 10–11 January 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 162–167. [CrossRef]
147. Agnew, D.; Del Aguila, A.; McNair, J. Enhanced Network Metric Prediction for Machine Learning-based Cyber Security of a Software-Defined UAV Relay Network. *IEEE Access* **2024**, *12*, 54202–54219. [CrossRef]
148. Pu, C.; Choo, K.K.R.; Korać, D. A Lightweight and Anonymous Application-Aware Authentication and Key Agreement Protocol for the Internet of Drones. *IEEE Internet Things J.* **2024**, *11*, 19790–19803.
149. Pu, C.; Warner, C.; Choo, K.K.R.; Lim, S.; Ahmed, I. liteGAP: Lightweight Group Authentication Protocol for Internet of Drones Systems. *IEEE Trans. Veh. Technol.* **2023**, *73*, 5849–5860. [CrossRef]
150. Famili, A.; Stavrou, A.; Wang, H.; Park, J.M.; Gerdes, R. Securing your airspace: Detection of drones trespassing protected areas. *Sensors* **2024**, *24*, 2028. [CrossRef]
151. Abir, M.A.B.S.; Chowdhury, M.Z.; Jang, Y.M. Software-defined uav networks for 6g systems: Requirements, opportunities, emerging techniques, challenges, and research directions. *IEEE Open J. Commun. Soc.* **2023**, *4*, 2487–2547. [CrossRef]
152. Shoufan, A.; AlNoon, H.; Baek, J. Secure communication in civil drones. In Proceedings of the Information Systems Security and Privacy: 1st International Conference, ICISSP 2015, Angers, France, 9–11 February 2015; Revised Selected Papers 1; Springer: Berlin/Heidelberg, Germany, 2015; pp. 177–195. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Article

Significance of Machine Learning-Driven Algorithms for Effective Discrimination of DDoS Traffic Within IoT Systems

Mohammed N. Alenezi

Computer Science & Information Systems Department, The Public Authority for Applied Education and Training, Safat 13147, Kuwait; mn.alenezi@paaet.edu.kw

Abstract: As digital infrastructure continues to expand, networks, web services, and Internet of Things (IoT) devices become increasingly vulnerable to distributed denial of service (DDoS) attacks. Remarkably, IoT devices have become attracted to DDoS attacks due to their common deployment and limited applied security measures. Therefore, attackers take advantage of the growing number of unsecured IoT devices to reflect massive traffic that overwhelms networks and disrupts necessary services, making protection of IoT devices against DDoS attacks a major concern for organizations and administrators. In this paper, the effectiveness of supervised machine learning (ML) classification and deep learning (DL) algorithms in detecting DDoS attacks on IoT networks was investigated by conducting an extensive analysis of network traffic dataset (legitimate and malicious). The performance of the models and data quality improved when emphasizing the impact of feature selection and data pre-processing approaches. Five machine learning models were evaluated by utilizing the Edge-IIoTset dataset: Random Forest (RF), Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and K-Nearest Neighbors (KNN) with multiple K values, and Convolutional Neural Network (CNN). Findings revealed that the RF model outperformed other models by delivering optimal detection speed and remarkable performance across all evaluation metrics, while KNN (K = 7) emerged as the most efficient model in terms of training time.

Keywords: artificial intelligence; machine learning; DDoS; deep learning; network security; detection; IoT security

1. Introduction

In today's rapidly digital interconnected world, cybersecurity stands as a critical cornerstone of organizational and governmental defense against various growing threats, ranging from data breaches to network attacks. Among these threats are distributed denial of service (DDoS) attacks which form a critical security challenge in modern networks, mostly targeting IoT devices [1,2]. Professionals in cybersecurity show increasing concern about the growing incidence and effects of DDoS attacks. The authors in [3] indicated that the number of major DDoS attacks rose significantly by 288% between 2021 and 2022. Additionally, defenses are becoming more complicated as attackers shift their focus from network infrastructure to application layers [4]. In addition, the growing number of IoT devices greatly increases the possible attack surface, thereby making the IoT more vulnerable to such attacks [5]. Cloudflare [6] claimed the prevention of over 21.3 million DDoS attempts in 2024, a 53% increase over 2023 with an average of 4870 attacks per hour. An extraordinary 5.6 Tbps attack around Halloween [6] brought attention to the serious danger that DDoS attacks represent, particularly given their capacity to interfere with

essential web services and cause serious operational and financial harm. In the financial sector, DDoS attacks have the potential to cause significant operational delays and financial losses [7]. These attacks can result in reputation damage in addition to the financial losses, especially for banks. The effects are not limited to the main target in the attack; they also encompass shared infrastructure, cloud providers, and energy usage [8]. The rapid increase in IoT devices raises cybersecurity risks in many areas and particularly for financial institutions, making them more exposed to attacks [9]. Devices such as cameras, sensors, and door controls are vulnerable, as attackers after compromising these devices could interrupt daily operations or gain access to sensitive information [10].

The IoT consists of interconnected devices that are capable of sharing data over the internet. The IoT network is undergoing an exponential growth with an increase from 30 billion devices in 2021 to an expectation of more than 75 billion in 2025 [11]. IoT devices vary widely in design and purpose, encompassing everything from smart household appliances and thermostats to industrial sensors and autonomous vehicles. Although the interconnectivity between IoT devices enhances data gathering and system operations, it introduces significant security vulnerabilities. Of particular concern is DDoS attacks, where attackers exploit vulnerable IoT devices to create large botnets capable of overwhelming network resources and disrupting major and critical services [1,12]. These disruptions could have an impact on many services, particularly in the health and financial sectors. Within the health sector, these attacks could disrupt critical operations, impact patient monitoring and decision-making in healthcare IoT environments [13]. Similarly, any interruption of services in the financial sector leads to a negative effect on the stock market and potential losses of thousands of dollars for every hour of downtime [14].

The detection and classification of DDoS attacks face several complex challenges in modern network environments. The effectiveness of the traditional signature-based detection methods has been reduced due to the adoption of traffic encryption [15]. Furthermore, the continuous evolution in attack patterns created by attackers to circumvent detection mechanisms creates ongoing challenges for security professionals. Additionally, the significant growth in legitimate traffic volume is creating additional complexity and challenges in differentiating between legitimate and attack traffic [15]. Studies by [16,17] show that the increase in IoT device usage has introduced new attack vectors that traditional detection systems struggle to discover effectively.

DDoS attacks present a real threat to IoT environments due to their inherent vulnerabilities, including limited memory and processing capacity. The complexity and magnitude of modern DDoS attacks are not adequately addressed by conventional security measures [15]. Cyber threat detection and mitigation may be enhanced by using machine learning (ML) solutions, such as supervised and deep ML algorithms, which can handle enormous datasets, discover complex patterns, and provide adaptive detection [18]. The effectiveness of these ML algorithms in detecting DDoS attacks in IoT systems should be thoroughly investigated. Although earlier studies have looked at ML techniques, they have not focused on the importance of feature selection when it comes to a model's effectiveness. Moreover, most previous efforts used accuracy as the primary focus to show a model's efficacy neglecting factors such as false-positive rates and training/detection time. Taking all of these factors into account provides insight on the optimal detection strategies to employ, given the limited resources and scalability of IoT devices. As a result, the choice of the appropriate ML models for network security in general and DDoS attacks in particular has become crucial in addressing these challenges.

Different ML and DL models showed capabilities in handling network traffic characteristics, processing speed, and detection accuracy rates. Recent studies [19,20] showed that the selection of the model can highly affect the detection accuracy rate up to approximately

25% and the processing speed by up to 69–76% under similar conditions in various domains. Furthermore, feature selection techniques effectively minimize training time and enhance the accuracy of ML-based detection [20,21]. The efficacy of different ML models varies noticeably, with some models showing superior performance in detecting certain attack types while struggling with other types.

Using the Edge-IIoTset dataset [22], this paper aims to thoroughly evaluate the effectiveness of different supervised ML classification algorithms to enhance detection of DDoS attacks in IoT environments.The most accurate and efficient models for attack detection is recommended by analyzing the computational training and detection time in addition to the performance metrics comparison.

The research methodology employs a systematic comparative analysis using the Edge-IIoTset dataset to assess multiple ML classification algorithms. The experimental framework covers data pre-processing, feature selection, and model training phases. The evaluated classification models are: LSTM, RF, CNN, KNN with varying k values (3, 5, 7), and SVM. Performance evaluation is carried out through statistical analysis of standard metrics including confusion matrix, while training time and detection speed are measured to evaluate computational efficiency.

The primary contributions of this research include:

- Presenting a detailed comparison of supervised ML and DL techniques in DDoS attack detection on IoT devices using the Edge-IIoTset dataset. Designed for IoT and Industrial Internet of Things (IIoT), this dataset provides a realistic and comprehensive cybersecurity resource.
- Investigating the important effect of feature selection on the accuracy and efficiency of detection models, an area that previous studies have not sufficiently emphasized.
- Evaluating several useful performance indicators, such as false-positive rates, training time, detection accuracy, and detection speed.
- Identifying the best algorithms for resource-constrained IoT devices through practical suggestions that take computational efficiency and performance into account.

The rest of this paper is structured as follows. The generic detection approaches and related ML algorithms are introduced in Section 2. Then the related work is presented in Section 3. In Section 4, the implementation details are explained including: data engineering and preparation framework, model selection and training details, and model evaluation. The results and model performance metrics are analyzed in Section 5. Finally, Section 6 outlines the study's limitations and future research directions, while the conclusion is presented in Section 7.

2. Background

2.1. Generic Detection Approaches

Network traffic analysis approach is one of the most basic ways to protect against DDoS attacks that target IoT devices. The network traffic is monitored for unusual patterns that indicate an ongoing attack. Organizations typically employ this method by integrating intrusion detection systems with other security measures that are intended to detect suspicious network behaviors [23]. These systems create baseline traffic patterns and detect substantial variations, allowing security teams to take action before services degrade. Even though it is widely used, network traffic analysis has some major problems especially when incorporated in IoT environments. Monitoring systems in IoT environments face challenges and difficulties when dealing with massive traffic, resulting in performance degradation and gaps in coverage [24]. Additionally, these methods have trouble handling encrypted communication, which hides packet contents and prevents careful examination. A study by [25] found that detection accuracy drops during flash crowds or seasonal traffic spikes,

leading to false alarms. Moreover, comprehensive monitoring needs expensive technology and experienced staff to evaluate findings, which can be costly for organizations with limited security resources.

Device monitoring strategies are also crucial for overall security as they focus on individual devices. Analyzing device logs and performance data can reveal specific device behaviors and possible compromise signs [26]. Important factors in reliable security systems include patterns of resource use, communication frequencies, and access scheduling. Reviewing these parameters together aids in locating infected devices displaying non-standard behavior, allowing for focused removal and repair. The IoT landscapes provide significant challenges for the implementation of device monitoring solutions. Many IoT devices with limited computational and storage capacity restrict logging and self-monitoring capabilities [27]. This results in unequal visibility over device populations: some give detailed data while others give little information. Additionally, closed systems and proprietary protocols that resist standardized monitoring complicate data collection even further. Kumar et al. [28] found that when the device populations rise into hundreds of millions of endpoints, the centralized collection systems would be overloaded and would cause delays in attack detection. Lastly, privacy concerns add even another level of complication as thorough monitoring may gather private user data, therefore generating problems with regulatory compliance.

Signature-based detection is among the first techniques used in networks to detect DDoS attacks. This method relies on known patterns of identified attack vectors, which makes it easier to identify them quickly when matching traffic patterns are present [29]. The implementation requires signature databases with unique byte sequences, packet structures, and connection patterns related to attack methods. The system initiates alarms and implements protective protocols upon the detection of incoming traffic corresponding to specific signatures. Although signature-based methods can quickly identify established attack methods, they have major drawbacks in IoT environments. First, signature-based approaches are ineffective against new or updated attacks not in existing signature databases [30]. Second, signature development lags behind attacker innovation as DDoS attacks develop, creating vulnerability windows between attack emergence and signature deployment. Third, maintenance requirements add additional burden as signature databases need constant improvements. Studies by [31] show that in IoT environments with restricted resources, signature-based solutions generate notable processing overhead, hence possibly influencing operational efficiency. Lastly, large-scale installations may result in too many false positives when the signatures are not detailed enough. The alarms may exhaust security workers and disguise serious dangers.

Machine learning classification approaches have been utilized for detecting DDoS attacks on IoT systems. Unlike traditional techniques, ML algorithms are more precise when it comes to distinguishing traffic attacks from flash crowd, thus lowering the false positive occurrence. In order to identify DDoS attacks in IoT environments, researchers have studied several supervised learning algorithms categorized as either conventional classification methods or DL approaches [32]. Conventional approaches set threshold values to identify attack traffic using extracted properties such as packet size, protocol type, behavior patterns. The utilization of AI-driven learning approaches have played a significant effect on the automatic generation of hierarchical representations from network data [33]. These techniques have special qualities; some perform well in identifying trends in sequential data and others may extract features from traffic representations that would otherwise be hidden in many dimensions.

In IoT environments, ML classification offers several advantages over traditional detection methods. Unlike signature-based systems limited to known attack patterns, classification models allow for the identification of new attacks displaying traits comparable to training samples [34]. Their multi-dimensional analytic features help to evaluate different traffic conditions, thereby strengthening the detecting systems. With sophisticated models lowering false positive rates by 40–60% while keeping detection sensitivity, research by [35] showed considerable increases in detection accuracy. The improved accuracy helps security specialists focus their attention on real dangers and saves their effort. Moreover, the flexibility of such systems proves to be an operational benefit as regular retraining solves evolving usage patterns and growing device populations [36]. Further benefits come from implementation flexibility as these models may be used depending on resource availability throughout several network tiers. Combining ML approaches with current security architecture improves protection and produces defensive systems that are more resistant to sophisticated attacks. Depending on the environment of application and particular requirements, ML techniques show different efficiency. Network designs, device populations, and attack strategies affect performance [37]. This diversity emphasizes the need for ongoing research targeted at the identification of ideal algorithms fit for certain operating environments and danger settings. Recent research has focused on developing frameworks for algorithm selection that takes predicted risks, network properties, and resource availability into account. The rapid development of IoT environments necessitates continuous research on flexible security solutions that efficiently handle new attack strategies and take into account the resource limitations common in IoT systems.

2.2. Machine Learning Models

Recurrent neural networks (RNN) were inspired by the biological human brain and are well-suited for many tasks. Unlike standard neural networks (feed-forward neural network), RNNs contain loops that provide short-term memory, enabling them to use previous inputs to shape current and future outputs. LSTM [38] networks are a type of RNN capable of capturing long-term dependencies. LSTM units contain specialized memory cells combined with standard recurrent units, enabling the network to remember information for an extended period of time. The information stored within these cells is managed by gates, including the input, output, and forget gates [38,39].

Support vector machine (SVM) is a supervised learning machine model developed by Valdimir et al. [40] in the 1990s. It is commonly used for classification and regression. By determining the optimal hyperplane, the model effectively splits data into distinct categories while maximizing the gap between classes. SVM initially only supported linear classification, then it evolved to non-linear classification problems. Linear SVMs split using a straight line. Nonlinear SVMs classify more complicated data patterns. SVMs are utilized in machine learning in many fields, such as text classification, spam identification, and prediction [40].

K-nearest neighbors (KNN) is a supervised learning algorithm that classifies or predicts values based on similarity to existing examples. Unlike other algorithms that build models while training, KNNs save data and decide what to do when necessary by finding the K training examples that are closest to a new point and voting or taking the average of those. The choice of distance measure such as Hassanat, Euclidean, and Manhattan significantly affects the performance. KNN works best when data are properly scaled, the right distance measure is selected, and an appropriate K value is chosen. Though KNN simplifies the processing of complicated patterns, KNN maintains high precision even in the event of up to 90% noise in the data [41,42].

Random forest is a supervised learning technique that uses many decision trees working together to make predictions. Designed by Leo Breiman in 2001 [43], it generates many trees using random data points and features, then aggregates the results. It is capable of working efficiently with complex data and relationships, requires minimal modification, accommodates missing data, and identifies the most critical variables. The method effectively identifies DDoS attacks by distinguishing malicious from legitimate network activity and evaluating anomalous traffic patterns. Combining several trees improves the algorithm's reliability by preventing overfitting and improving prediction accuracy on new data [43–46].

Convolutional neural networks (CNNs) are a subset of DL architectures, developed in the 1980s by Yann LeCun [47]; these networks extract hierarchical characteristics from incoming data by performing convolution operations through a number of processing layers. CNNs were originally inspired by biological visual processing systems; however, they have expanded their capabilities beyond picture classification to include text, audio, and network traffic classification. CNNs instantly learn the best way to describe features straight from data, which makes classification more accurate across a wide range of domains. This is in contrast to traditional ML techniques that require manual feature engineering. Due to their capacity for self-learning, sharing of parameters, and preservation of spatial correlations, CNNs have emerged as powerful AI tools for pattern recognition and classification tasks [48,49].

3. Related Work

IoT has rapidly expanded to link many systems and devices. With this linkage comes security concerns that must be addressed. Researchers [50–52] have proposed taxonomies to classify IoT risks into several levels: service, device, infrastructure, and communication. IoT systems' physical objects, protocols, data, and software components are all open to security breaches [53]. To handle these problems, researchers have proposed several countermeasures including key managements, authentication, access control, and privacy preservation strategies [51,54,55]. They also focused on developing robust detection methods capable of identifying attack signatures within the unique constraints of IoT environments [50,56]. However, the protection of IoT ecosystems remains a challenge due to device limitations and non-standard IoT settings [51,57,58].

Agent-based approaches are being investigated recently to reduce DDoS attacks on IoT systems. The researchers in [59,60] explored and studied adaptive traffic filtering, anomaly detection using ML techniques, and collaborative agent-based detection systems. Some researchers have built agent-based simulators to study protection measures against DDoS attacks [61], and other suggest lightweight agents for IoT to identify and mitigate attacks [60,62]. As novel techniques, blockchain-based collaborative detection and learning-driven detection mitigation have been investigated [63–65]. Although these approaches improve IoT network resilience against evolving DDoS attacks while utilizing scalability and resource limitations, they still present some drawbacks. Agent-based techniques commonly face challenges related to resource consumption when deployed on resource-limited IoT devices, potentially creating performance bottlenecks when deployed at scale. Collaborative detection systems, while effective in theory, face significant challenges with secure communication between agents and can fail when attackers target the coordination mechanisms themselves. Approaches like adaptive traffic filtering, blockchain-based detection, and learning-driven mitigation suffer from high false positive rates when facing sophisticated low-rate attacks, and typically require extensive training data that may not be representative of evolving attack patterns.

Ferrag et al. [22] presented Edge-IIoTset, a complete cybersecurity dataset for IoT and IIoT applications, which includes realistic network traffic from over 10 types of IoT devices across a seven-layer testbed architecture, thereby addressing constraints in existing datasets. Deep neural networks (DNNs) in centralized and federated learning environments, as well as traditional ML techniques like DT, RF, SVMs, and K-Nearest Neighbor (KNN) were evaluated. Accuracy values of 99.99% for binary classification and 94.67% for 15-class classification were obtained by using DNNs. The dataset outperforms current IoT/IIoT datasets and offers privacy-preserving federated learning, making it a useful resource for building intrusion detection systems in IoT/IIoT environments.

Designed especially for intrusion detection in IIoT systems, Rashid et al. [66] presented a federated learning architecture. Their method uses CNNs and RNNs as baseline classifiers inside a federated learning system. The system is evaluated on the Edge-IIoT dataset and achieved an accuracy of 92.49%. Although it is less accurate than centralized models (93.92%), the suggested architecture offers significant advantages by lowering bandwidth use and ensuring data privacy. The key contribution of this study is the capacity of peripheral devices to perform intrusion detection independently, free from the need of always being connected to a central server. This immediately tackles IoT network's basic issues with limited bandwidth and privacy.

Focusing on DenseNet and Inception Time architectures across three main datasets: ToN-IoT, UNSW-NB15, and Edge-IIoT, the authors in [67] performed a detailed comparative evaluation of DL models for IoT cybersecurity. The authors modified DenseNet for one-dimensional input vectors and used sliding window methods with Inception Time to enhance temporal feature extraction. Inception Time performed remarkably, achieving 100% accuracy on the ToN-IoT dataset, 94.9% on Edge-IIoT, and 98.6% on UNSW-NB15. These results show the remarkable effectiveness of the Inception Time architecture.

Hnamte and Hussain [68] proposed a hybrid DL architecture that combines Deep Convolutional Neural Networks (DCNN) with Bidirectional Long Short-Term Memory (BiLSTM) networks. The proposed approach catches temporal correlations and spatial features by combining BiLSTM components with CNN layers. The authors claim that the model has a low false positive rate, with an accuracy of 100% on the CICIDS2018 dataset and 99.64% on the Edge-IIoT dataset. These results therefore show the benefits of hybrid designs over single-model methods.

Khacha et al. [69] developed a tailored hybrid DL model that combines CNN and LSTM architectures. The CNN-LSTM model achieved 100% accuracy in binary classification and performed well in multi-class scenarios compared to both traditional ML models and individual DL models. Thiyam and Dey [70] tackled the challenge of class imbalance in cybersecurity datasets by adopting a unique feature assessment approach. The approach combined feature shuffling methods using RF for optimal feature selection with a hybrid resampling technique integrating SMOTE (Synthetic Minority Over-sampling Technique) and TOMEK link. Using six ML techniques on the Edge-IIoT and CIC-DDoS2019 datasets, the decision tree (DT) classifiers achieved respective accuracies of 99.32% and 99.87%, respectively. This shows rather clearly how fixing class imbalance improves detection performance. Table 1 highlights studies using the Edge-IIoT dataset for multi-class classification, presenting the Edge-IIoT results from studies that evaluated multiple datasets.

Table 1. Summary of reviewed studies.

Research	Year	Dataset	Model	Accuracy	Performance Metrics	ML	DL	IoT	Approach
Ferrag et al. [22]	2022	Edge-IIoTset	DT, RF, SVM, KNN, DNN	94.67–96.00% †	P, R, F1, DT, TT	✓	✓	✓	CL + FL
Rashid et al. [66]	2023	Edge-IIoTset	CNN, RNN	92.49%	P, R, F1	✗	✓	✓	FL
Khacha et al. [69]	2023	Edge-IIoTset	CNN-LSTM	98.69%	P	✗	✓	✓	Hybrid DL + CL
Tareq et al. [67]	2022	Edge-IIoTset	Inception Time	94.94%	P, R, F1	✗	✓	✓	DL Architecture + CL
Hnamte et al. [68]	2023	Edge-IIoTset	DCNNBiLSTM, DNN, CNN, LSTM, AE	99.64%	DT, TT	✗	✓	✓	Hybrid DL + CL
Thiyam et al. [70]	2023	Edge-IIoTset	KNN, LR, DT, RF, ADB, GDB	99.32% *	P, R, F1	✓	✗	✓	FE + CL
Present Work	2025	Edge-IIoTset	RF, KNN, SVM, LSTM, CNN	99.99%	P, R, F1, DT, TT	✓	✓	✓	FE + CL

P: Precision; R: Recall; F1: F1-Score; DT: Detection Time; TT: Training Time; ML: Machine Learning; DL: Deep Learning; IoT: Internet of Things; FL: Federated Learning; FE: Feature Engineering; CL: Centralized Learning; Performance Metrics: Available performance and timing metrics. † Multi-class accuracy ranges shown where applicable. * Binary classification.

4. Materials and Methods

Presented in this section is the methodological framework for the implementation and evaluation of DDoS detection algorithms in IoT systems. This research assesses the efficacy of a variety of ML techniques, including KNN with varying K values, RF, SVM, LSTM, and CNN, in identifying DDoS attacks within IoT traffic patterns. Three primary phases define the approach: performance evaluation; model selection and training; data engineering and preparation. To extract differentiating attributes and address class imbalance issues, the Edge-IIoTset dataset [22] is investigated during the data preparation phase. Configuring the selected ML algorithms with appropriate architectural parameters and implementing training protocols that are optimized for IoT constraints are carried out during the model development phase. This research evaluates algorithm performance using criteria such as performance metrics (detection accuracy, precision, recall, F1-score), computational efficiency, and the speed of detection.

4.1. Data Engineering and Preparation Framework

4.1.1. Dataset Acquisition and Cleaning

The selection of data is a vital step in developing any ML and DL model. It is a highly important stage in the construction of a robust DDoS attack detection model, since every model requires proper training and validation on sufficiently large datasets that are free from noise, outliers, missing data, and so on. With the right dataset selected for training, an ML model would be able to identify unseen traffic patterns more accurately. The Edge-IIoTset dataset is a popular Kaggle dataset specifically created for DDoS attack identification utilizing both ML and DL frameworks [22]. The Edge–IIoTset corpus is made up of 49 data files that are organized into three distinct sub-directories: regular IoT and IoT application traffic, malicious IoT and IoT application traffic, and a specific dataset used for ML and DL.

Data cleaning is the act of fixing problems like outliers and missing numbers in order to prepare a dataset for analysis. There are several reasons why there could be missing values, including lost packets or insufficient logging. Methods such as mean/median imputation, forward/backward fill, or deletion are used to deal with missing values. The identification and elimination of outliers is also essential since they can impact the model's

perception of legitimate versus malicious traffic. This process also removes unwanted and useless information, providing data columns to make it more suitable for analysis.

4.1.2. Data Transformation

Data transformation refers to the encoding and normalization of the data parameters or features to make them suitable for the model. Encoding transforms the categorical parameters into a numerical format to make it easy for the models to interpret. For the purpose of this research, two types of encoding are used:

- Label encoding: It directly converts categorical values with numerical substitutes.
- One-hot encoding: It ensures that no ordinal associations are identified by the model through converting categorical variables into a sequence of binary columns. Each category is encoded by a binary vector with just one member assigned the value '1' and the remaining components set to '0'.

Each parameter has a different range of values. So, it is necessary to make all the parameter values follow a common range. Normalization is normally used to scale up the numerical values to a common range. Min–max normalization is mainly used in this study.

4.1.3. Feature Extraction and Selection

In developing an effective ML model, proper pre-processing of data should be conducted. This pre-processing includes feature selection and extraction, which, in turn, transforms raw network data into structured representations that are suitable for classification algorithms. There are several network attributes that can be utilized in DDoS detection such as packet size distributions, flow duration metrics, protocol type indicators, TCP flag configurations, and network addressing parameters (source/destination IP addresses and ports). The selection method uses both a statistical technique as well as domain knowledge to extract the most informative attributes and eliminate redundant features. This approach enhances computational efficiency without compromising classification performance, as demonstrated by feature selection improving RF training time from 21.028 s to 16.455 s while increasing accuracy from 93.20% to 99.99%, and reducing SVM training time from 334.456 s to 85.322 s while improving accuracy from 75.06% to 99.26% (24.20% improvement). In addition to computational efficiency, this approach improves detection time and other performance criteria. Thorough analysis and sufficient quantitative evidence supporting these improvements are presented in Section 5. In this research, principal component analysis (PCA) was applied to minimize dimensionality and transform potentially correlated features into linearly uncorrelated variables ordered by explained variance. This method ensures the preserve of essential information while reducing computational requirements.

4.1.4. Data Splitting and Imbalanced Classes

The whole dataset is split into three independent datasets: a training set, validation set, and testing set. In total, 80% of the data corpus is used to train the model and 20% is used for model evaluation and testing. The training dataset is also split into training and validation sets using an 80:20 ratio. The training dataset is used to train the model and the validation set is used to tune the hyper-parameters and avoid overfitting. Data splitting might showcase the existence of imbalanced classes, and this can be dealt with by using either undersampling or oversampling. Class weights were assigned for each class throughout the training phase of this study in order to handle unbalanced data. The detailed class distribution demonstrating this imbalance is provided in Appendix A.

Real-time threat identification capabilities are an important aspect of IoT DDoS detection. Detection time is an essential performance parameter, alongside accuracy, in IoT

environments, since these environments are characterized by devices with limited processing capacity and resources. Consequently, to ensure practical deployment viability in real-world IoT scenarios, where immediate threat response is crucial for preventing service disruption and network compromise, both high accuracy and quick detection are established as essential requirements.

4.2. Model Selection and Training

4.2.1. Model Selection

While DL methods are traditionally associated with image and sequential data, their application to network traffic analysis is well-justified by the inherent characteristics of DDoS attack patterns and IoT network flows. Sequential connectivity and temporal interconnections in network traffic data demand advanced pattern recognition capabilities. CNNs can capture local feature patterns and complex interactions inside network flow characteristics by considering feature vectors as 1D sequences, which enables the identification of attack signatures across multiple feature dimensions simultaneously. The comprehensive evaluation of both traditional ML and DL methods overcomes the variety of computing needs and deployment scenarios existing in IoT systems. For IoT systems with limited resources, this makes it possible to create hybrid frameworks that combine the computational efficiency of conventional techniques with the advanced pattern recognition abilities of DL. LSTMs are particularly suitable for this domain as DDoS attacks exhibit sequential dependencies and temporal behavioral patterns that traditional ML methods could not totally capture. In this work, five distinct ML and DL models are evaluated: RF, SVM, KNN with varying k values, LSTM, and CNN.

4.2.2. Model Architecture

- RF: the RF classifier is created with 100 trees in the forest. The maximum depth for each tree was not set, allowing it to grow until all the leaves were pure or the node contained insufficient samples for splitting. The model parameters specified two samples minimum for internal node splitting and one sample minimum per leaf. These settings were chosen for the purpose of balancing the models complexity and its ability to generalize well.
- SVM: it uses a regularization parameter (C) of 1.0 to balance low error on the training data and weight norm optimization. A linear kernel was used, making the decision boundary a straight line. The influence of a single training example is determined by the gamma parameter, which was set to its default value of 1 /(number of features).
- KNN: the KNN model's algorithm type was left at auto to allow the model to choose the optimum method for the dataset's structure. To generate predictions, k values (3, 5, 7) of neighbors were tested. A uniform weight function was used so that all locations in each neighborhood weigh equally, regardless of their distance from the goal.
- LSTM: the LSTM networks' ability to learn and recall over extended sequences makes them ideal when it comes to finding patterns in time-series data connected with DDoS attacks. The sequential architecture employs four LSTM layers containing 50, 50, 100, and 100 units, respectively, interspersed with dropout layers for regularization, followed by a final dense layer with 15 output neurons for multi-class DDoS detection.
- CNN: the detailed structure is made up of convolutional, pooling, and fully connected layers that have been tailored to the dataset's specific attributes. CNNs are able to detect complex trends in network traffic data because they easily identify spatial hierarchies in the data. The model is made up of five one-dimensional convolutional layers, using 32, 64, 128, 128, and 128 filters, respectively. Each convolutional layer is followed by a max pooling layer to reduce the feature size. After these layers, the

output is flattened and passed through two fully connected (dense) layers: first with 64 neurons, then with 15 neurons, which produce the final classification.

4.2.3. Model Training

In ML and DL model training, data are fed into the model and continuously adjusted to minimize a loss function. DL uses multiple-layer neural networks along with very large datasets and substantial computing capacity to discover intricate patterns in the data. In this study, 80% of the entire data is used for training and validation. Throughout the training process, several strategies were used to prevent overfitting and help the models generalize appropriately. To improve generalization and avoid relying greatly on certain neurons, dropout layers were added to the neural networks in the DL models (CNN and LSTM). The dataset was carefully split into training, validation, and testing sets to make sure that the models were tested on new data. Traditional ML models such as RF, KNN, and SVM employed built-in regularization methods, and hyperparameter optimization was conducted to achieve a balance between model complexity and generalization performance. Cross-validation was applied when needed, to ensure robust model selection and parameter tuning, maintaining a reliable evaluation framework for practical IoT deployment scenarios.

Specific stopping criteria were defined for each model type to ensure optimal performance. Early stopping was used for DL models such as CNN and LSTM by monitoring validation accuracy and automatically terminating training when validation performance stopped improving. Traditional ML models implemented algorithm-specific stopping criteria: RF used a predetermined number of estimators, KNN classified using distance-based classification without iterative training, and SVM optimized using convergence tolerance parameters.

4.3. Model Evaluation

The efficacy of a trained model is evaluated using a variety of critical metrics, such as precision, recall, accuracy, and F1-score. Computational efficiency, including both training and detection time, is evaluated alongside confusion matrix analysis within the evaluation framework to ensure a comprehensive performance evaluation for IoT deployment scenarios. In this way, not only classification accuracy is evaluated, but also the practical feasibility of implementing these models in IoT environments with limited resources is thoroughly considered. Typically, this approach includes an unbiased evaluation of the model's performance using an independent test dataset that was not utilized for training. In general, a model's evaluation helps determine its performance in real scenarios and provides direction for additional model selection or adjustments. All the testing and model evaluations were carried out on a Dell Precision WorkStation T7500, which featured two Intel Xeon X5570 processors running at 2.93 GHz, 64 GB of RAM, and Windows 10 Pro Education. Python 3.11.11 was used for all experiments.

5. Results and Analytical Discussion

The experimental investigation of ML and DL algorithms for the detection of DDoS attacks in IoT environments revealed significant variations in performance and computational trade-offs. In this section, thorough interpretations of the experimental results, with emphasis on implications for IoT implementations with restricted resources are presented.

5.1. Algorithm Performance Comparison

5.1.1. Classification Performance Metrics

A hierarchy of performance efficacy was revealed through the comparison of several categorization methods. With an accuracy of 99.99%, RF classifier (RFC) showed superior

performance than other techniques in the classification of DDoS and other attack types. The CNN model attained the second-highest accuracy with 99.80%; SVM followed with 99.26%. With KNN (K = 7) at 99.10%, KNN (K = 5) at 99.09%, and KNN (K = 3) at 99.14%, the KNN variations showed equivalent accuracy levels. LSTM networks obtained an accuracy of 99.25%.

The analysis of the confusion matrix revealed that RFC maintained consistent performance across all DDoS attack classes (DDoS_HTTP, DDoS_ICMP, DDoS_TCP, and DDoS_UDP), with minimal misclassifications. This insight is especially important considering the multi-class classification challenge, including 15 different kinds of attacks and consistent traffic patterns.

Figure 1 compares the performance metrics of all evaluated models, demonstrating the considerable overall accuracy advantage of RFC and CNN. With the 99.80% highest precision, RFC proved remarkable capacity to prevent false positives across all DDoS variants and other attack types. CNN showed 99.00% precision, same as KNN variants. The LSTM attained 97.00%, while the SVM had a precision of 99.00%. These metrics evaluate the capacity of each model to accurately identify DDoS attack without incorrectly classifying legitimate traffic.

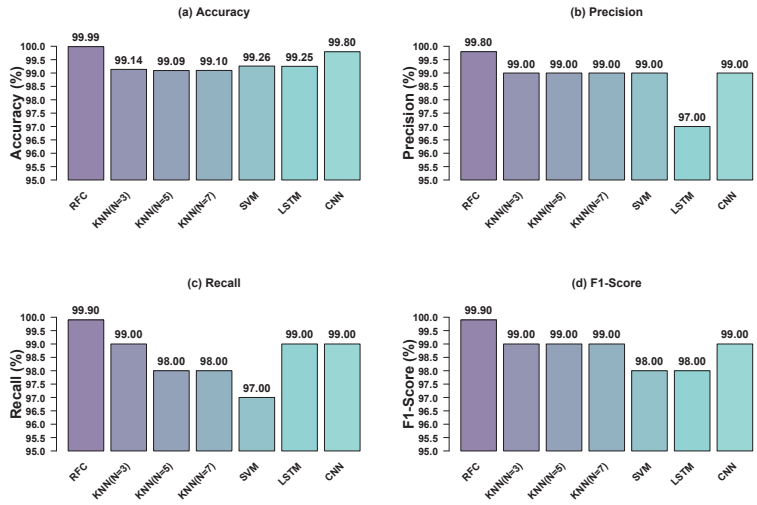


Figure 1. Comparison of performance metrics across all evaluated models.

RFC once again outperformed other models in terms of recall, achieving a remarkable 99.90% accuracy rate in identifying all DDoS attacks. CNN and LSTM both scored 99.00%, KNN (K = 3) scored 99.00%, and KNN = (K = 5, 7) scored 98%. With even lower recall at 97.00%, SVM found some undetectable attacks.

The F1-score followed similar trends with RFC scoring 99.90%, followed by CNN and KNN = (K = 3, 5, 7) all at 99.00%. With 98.00%, LSTM and SVM showed rather lower F1-scores. More details, including heatmaps and full confusion matrix visualizations, are provided in Appendix A.

5.1.2. Misclassification Analysis

Examining the confusion matrix closely produced important new understanding of misclassifications between several DDoS variations. Four different DDoS attack forms are included in the dataset: DDoS_HTTP, DDoS_ICMP, DDoS_TCP, and DDoS_UDP. Table 2 shows that classification performance for each DDoS type differed significantly between models in the base run. Note that all percentages results are rounded to one decimal place for clarity.

Table 2. Misclassification rates for DDoS attack variants across models.

Model	DDoS_HTTP	DDoS_ICMP	DDoS_TCP	DDoS_UDP
RFC	0.0%	0.04%	0.0%	0.0%
CNN	0.0%	0.0%	0.0%	0.41%
LSTM	0.0%	0.39%	0.0%	0.0%
KNN (K = 3)	26.3%	0.43%	21.4%	0.0%
KNN (K = 5)	26.8%	0.51%	19.8%	0.0%
KNN (K = 7)	27.2%	0.55%	19.5%	0.0%
SVM	39.4%	0.35%	43.1%	0.0%

With only one false negative noted across all DDoS classes, RFC showed a remarkable accuracy for all DDoS variations. Particularly, all DDoS_HTTP (2123 samples), DDoS_TCP (2075 samples), and DDoS_UDP (2890 samples) were correctly identified; DDoS_ICMP had only one misclassification out of 2554 samples. Additionally, RFC demonstrated remarkable non-DDoS attack detection with only four cases across 18,798 samples, including single misclassifications in Port_Scanning and Uploading categories, and two cases in XSS classification. These exceptionally high accuracy rates across multiple algorithms (RFC 99.99%, CNN 99.80%, SVM 99.26%) raise important considerations regarding potential overfitting to the Edge-IIoTset dataset characteristics. Such uniformly high performance may indicate that models have learned dataset-specific patterns rather than generalizable attack signatures, particularly given the controlled laboratory environment in which the data were collected. For most DDoS variations, CNN performed remarkably well; however, it showed particular flaws with DDoS_UDP, where 12 cases were misclassified as DDoS_ICMP. This misclassification pattern indicates that the network traffic patterns of these attack types, which challenged CNN's feature extraction capabilities, are similar.

The performance of KNN variants varied depending on the DDoS types. For DDoS_HTTP attacks, KNN (K = 3) misclassified 502 instances (23.6%) as XSS attacks, and 56 instances (2.6%) as Vulnerability_scanner. Similarly, KNN (K = 5) misclassified 504 cases (23.7%) as XSS and 64 instances (3.0%) as vulnerability scanner attacks. With 527 occurrences (24.8%) misclassified as XSS, and 51 instances (2.4%) as vulnerability scanner attacks, KNN (K = 7) displayed similar misclassification rates.

With misclassification rates of 19.9% (412 instances), 19.8% (410 instances), and 19.5% (404 instances), KNN variants K = 3, 5, and 7 misclassified DDoS_TCP attacks as Port_Scanning attacks, respectively. Remarkably, KNN (K = 3) also misclassified 28 DDoS_TCP instances (1.3%) as backdoor attacks, 3 instances (0.14%) as fingerprinting, and 1 instance (0.05%) as ransomware giving a total of 21.4% misclassification rate for KNN (K = 3). DDoS_ICMP exhibited rather good detection across the KNN versions, with misclassification rates below 1%, mostly as fingerprinting attacks (11, 13, 14 instances misclassified for K = 3, K = 5, and K = 7, respectively).

SVM showed different performance depending on the DDoS type. It performed poorly for DDoS_TCP with just 56.9% (1181 instances) correctly classified and a total misclassification rate of 43.1%, including 41.2% (855 instances) misclassified as Port_Scanning attacks and 1.9% (39 events) misclassified as ransomware attacks. For DDoS_ICMP, it successfully classified 99.65% of instances with only 0.35% misclassification. With 37.9% (804 instances) misclassified as XSS attacks and 1.5% (33 instances) as Vulnerability_scanner attacks, SVM precisely classified just 60.6% (1286 cases) for DDoS_HTTP, giving a total misclassification rate of 39.4%.

For most DDoS variations, LSTM performed well; however, it showed certain challenges with DDoS_ICMP, misclassifying 10 cases (0.39%) as DDoS_UDP attacks. This pattern is consistent with CNN's behavior and suggests that these attack categories share

inherent similarities, which present a challenge to both DL methods. Table 3 reflects the performance comparison of all evaluation metrics for ML/DL models.

Table 3. Performance evaluation of classification models.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
RFC	99.987	99.9	99.9	99.9
CNN	99.796	99.0	99.0	99.0
SVM	99.261	99.0	97.0	98.0
LSTM	99.251	97.0	99.0	98.0
KNN (K = 3)	99.136	99.0	99.0	99.0
KNN (K = 5)	99.093	99.0	98.0	99.0
KNN (K = 7)	99.097	99.0	98.0	99.0

5.2. Computational Efficiency Analysis

5.2.1. Detection Time Efficiency

Figure 2 displays the detecting time efficiency of every model. RFC, with a detection time of just 0.584 s, demonstrated remarkable operational efficiency, making it the most appropriate option for real-time IoT deployment scenarios requiring rapid DDoS attack detection. CNN followed with 4.121 s and LSTM needed 13.580 s for detection.

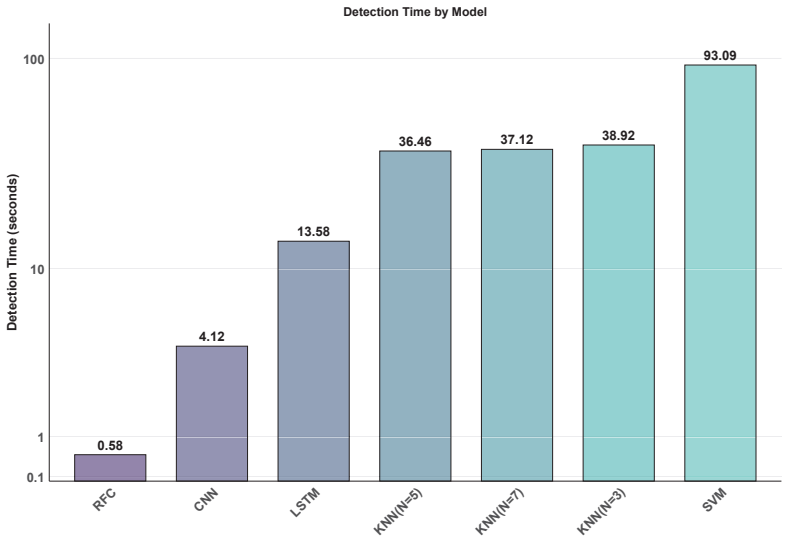


Figure 2. Detection time comparison across all models.

KNN variants required considerably more time, with K = 5 being the fastest at 36.457 s, followed by K = 7 (37.119 s) and K = 3 (38.923 s). SVM was the slowest model with 93.092 s for detection.

Regarding IoT deployment situations, these differences are important. IoT systems sometimes run under restrictions including limited processing capacity, power limits, and real-time decision-making requirements. Fast detection times of RFC make it especially appropriate for IoT setups with limited resources as quick DDoS attack detection is required to keep service availability.

5.2.2. Training Time Efficiency

The training times required for each model are presented in Figure 3. Among the KNN variations, KNN (K = 7) needed just 0.162 s, KNN (K = 3) needed 0.164 s, and KNN (K = 5) required 0.175 s, thus demonstrating remarkable training efficiency. RFC was also relatively efficient during the training phase, requiring 16.455 s. By contrast, the DL models

demanded considerably more computational resources during training. LSTM displayed the longest training time at 4986.316 s (about 83 min), while CNN required 1618.678 s (almost 27 min). Requiring 85.322 s, SVM positioned itself between traditional ML methods and DL approaches.

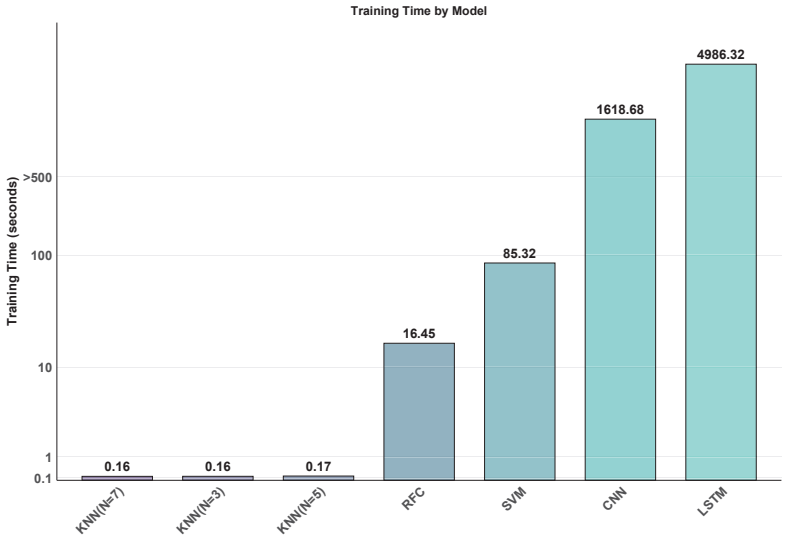


Figure 3. Training time comparison across all models.

These differences in training times highlight important considerations for the cycle of model development and application. While KNN variants offer rapid training, which is helpful in constrained computing environments or scenarios requiring frequent model updates, their detection performance for DDoS attacks remains weaker compared to RFC. Although the DL models impose heavy computational demands during training, making them less suitable for contexts with limited computational resources, they still achieve respectable accuracy in DDoS detection. Table 4 reflects the computational efficiency for all models with ranking based on their performance.

Table 4. Computational efficiency analysis of evaluated models: training and detection times with ranks.

Model	Training Time (s)	Detection Time (s)	Training Rank	Detection Rank
KNN (K = 7)	0.162	37.119	1	5
KNN (K = 3)	0.164	38.923	2	6
KNN (K = 5)	0.175	36.457	3	4
RFC	16.455	0.584	4	1
SVM	5.322	93.092	5	7
CNN	1618.678	4.121	6	2
LSTM	4986.316	13.580	7	3

5.3. Evaluating Feature Selection

5.3.1. Performance Improvements

The analysis of the base results compared to the enhanced results reveals significant improvements achieved through feature selection and data pre-processing approaches. Table 5 presents a comparison of the base versus the enhanced accuracy across all algorithms.

Table 5. Impact of feature selection on model accuracy.

Model	Accuracy (Base)	Accuracy (Enhanced)	Improvement	Detection Time Impact
RFC	93.20%	99.99%	+6.79%	+0.04 s
KNN (K = 3)	81.03%	99.14%	+18.11%	+1.87 s
KNN (K = 5)	80.92%	99.09%	+18.17%	+3.26 s
KNN (K = 7)	81.06%	99.10%	+18.04%	+4.46 s
SVM	75.06%	99.26%	+24.20%	+8.34 s
LSTM	98.83%	99.25%	+0.42%	+0.00 s
CNN	74.58%	99.80%	+25.22%	+0.00 s

The most notable improvements were observed in CNN, with a 25.22% increase in accuracy, and SVM with a 24.20% increase. All of the KNN variants experienced substantial improvements of approximately 18%. RFC exhibited a 6.79% improvement, while LSTM showed the smallest enhancement at 0.42%.

5.3.2. False Positive Reduction in DDoS Classification

For all models, feature selection especially affected false positive rates for DDoS attack types. The following DDoS specific classification errors were rather common in basic setups without optimal feature selection:

DDoS_HTTP attacks: KNN variants misclassified about 27% of DDoS_HTTP instances in base configuration; most were incorrectly labeled as XSS attacks. SVM initially misclassified 804 instances (37.9%) of DDoS_HTTP as XSS attacks. Even after feature improvement, KNN variants still showed flaws in DDoS_HTTP classification, although with much lower error rates (23.6–24.8%).

DDoS_TCP attacks: with misclassification rates of approximately 20% for KNN variants and 43% for SVM, these models often confused DDoS_TCP with Port_Scanning attacks in basic configuration. KNN variants still showed the same confusion pattern after feature improvement, but with lower rates (19.5–19.9%). For this attack type, SVM's improvement was negligible, with 41.2% of DDoS_TCP instances still misclassified as Port_Scanning after feature upgrade.

DDoS_ICMP attacks: for DDoS_ICMP detection, base configuration already showed rather high accuracy among the models. With misclassification rates below 1% for most models, feature improvement still managed to further raise detection rates. In particular, SVM showed improvement, lowering misclassifications to 0.4%.

DDoS_UDP attacks: detection of DDoS_UDP attacks was nearly flawless in both base and improved configurations across all models, indicating that this attack type has quite unique and easily recognizable traffic patterns.

5.3.3. Algorithm-Specific Feature Response Mechanisms

The varying responses to feature selection across algorithms can be attributed to their fundamental learning mechanisms when applied to DDoS detection. The performance of SVM is determined by the separation of hyperplanes within the feature space. It is particularly notable that enhanced feature selection succeeded in eliminating noisy features that had previously obstructed the optimal formation of hyperplanes between different DDoS variants, especially for DDoS_HTTP. However, despite the overall improvement, SVM continued to struggle with DDoS_TCP attacks, maintaining a high misclassification rate of 41.2% even after feature enhancement.

Convolutional operations in CNN greatly benefited from the provision of well-prepared input data. The improved feature selection enabled more distinguishable patterns

to be captured by CNN's convolutional layers, particularly facilitating the differentiation between DDoS_ICMP and DDoS_UDP attacks.

The operation of KNN algorithms fundamentally depends on distance calculations within the feature space. Feature selection effectively reduced dimensionality while preserving critical discriminative features, thereby allowing for more accurate nearest-neighbor identification among DDoS variants. However, KNN continued to struggle in distinguishing certain attack types (such as DDoS_HTTP versus XSS, and DDoS_TCP versus Port_Scanning), suggesting that inherent similarities in their traffic patterns challenge distance-based classification methods.

RF showed moderate improvement, suggesting its ensemble nature and internal feature selection mechanisms already captured significant discriminative information for DDoS attack variants even in the base configuration. LSTM's minimal improvement suggests its sequential learning capabilities were already effectively capturing temporal patterns in DDoS attack traffic.

The observed variations in performance point to the fundamental limitations of each algorithm when handling complex attack signatures in IoT systems. Enhanced feature selection can improve overall results, but ongoing misclassification across different methods shows that it remains challenging to distinguish between attacks with similar characteristics. These trends emphasize the importance of thoroughly understanding the feature space when choosing detection methods that are best suited for specific IoT deployment scenarios.

5.4. Discussion and Methodological Considerations

5.4.1. Experimental Refinements and Performance Assessment

The initial findings, which revealed nearly perfect accuracy for certain models, required careful review especially with regard to DDoS attack detection. The confusion matrix data gave more consistent and thorough insights by showing particular errors in classifying different types of DDoS. This detailed study identified flaws that were not obvious in aggregate measures, emphasizing the importance of going beyond surface-level performance data for a more thorough examination. For example, even with CNN's general accuracy of 99.80%, the confusion matrix found specific challenges in differentiating DDoS_ICMP from DDoS_UPD attacks. However, these optimized results must be interpreted with caution regarding real-world deployability. The consistent high performance across diverse algorithms suggests possible dataset-specific optimization that may not translate to operational IoT environments, where traffic exhibits greater variability, noise, and unpredictable device behaviors than captured in controlled datasets. KNN variants similarly struggled particular DDoS kinds even with their apparently high overall accuracy. This result is especially important, since it shows how great overall accuracy could hide significant misclassifications at the level of specific attack type. Therefore, it becomes essential to evaluate model performance not only by means of general indicators but also by closely analyzing behavior over several attack types.

The final DDoS detection results were much influenced by several experimental changes:

Feature normalization: particularly for SVM and DL models, the change from standard scaling to min–max scaling clearly enhanced model performance for DDoS detection. Comparatively to multiple misclassifications seen with conventional scaling, the confusion matrix for RF with min–max scaling revealed flawless categorization for all DDoS_HTTP, DDoS_TCP, and DDoS_UDP cases.

Hyperparameter optimization: models such as LSTM gain from well tuned hyperparameters for DDoS detection including learning rate scheduling with early stopping. By use of early stopping, the LSTM model demonstrated convergence at epoch 44, therefore preventing overfitting and optimizing performance.

Architectural refinements: in CNN's architecture, five convolutional layers with max-pooling operations helped hierarchical feature extraction, which proved especially successful for different DDoS attack variants with subtle variations in network traffic patterns.

5.4.2. Notable Classification Patterns in DDoS Detection

The evaluation of DDoS attack detection techniques highlighted several interesting trends:

1. After adjusting feature selection, there was little performance difference between KNN variants, regardless of K value, suggesting that neighborhood size had little impact on DDoS classification.
2. The LSTM showed a good initial performance in the detection of DDoS_UPD and DDoS_TCP even in the lack of improved feature selection. This observation implies that its temporal learning capacity sufficiently caught the sequential patterns defining these attack forms.
3. RF's remarkable combination of accuracy (99.99%) and detection efficiency (0.584 s) underscores the strength of ensemble-based techniques for tackling difficult multi-class issues with many attack variants.
4. The improvement in LSTM's performance following feature addition was negligible (only 0.42%), in sharp contrast to the significant gains noted in other models. This finding suggests that the recurrent architecture of LSTMs unintentionally covers relevant temporal connections in network traffic data, therefore reducing the benefits possible by feature selection.
5. With misclassification rates ranging from 0% (RF) to as high as 39.4% (SVM, with 37.9% specifically misclassified as XSS attacks), the DDoS_HTTP attack type displayed clearly notable variations in categorization accuracy among models. This variation suggests that this particular DDoS variant shows complex traffic patterns, which may be especially challenging for methods depending on hyperplane-based or distance-based decision boundaries.

5.4.3. Attack Variants and Feature Representations

This research makes use of the Edge-IIoTset dataset comprising four main DDoS attack variants: DDoS_HTTP, DDoS_ICMP, DDoS_TCP, and DDoS_UDP. Each variant includes distinct network traffic patterns and exploitation techniques that significantly affect detection performance among the evaluated models. Typically reflecting real-world traffic behavior, DDoS_HTTP attacks overwhelm web servers with HTTP requests. The tendency of KNN variants to misclassify a significant fraction of DDoS_HTTP events as XSS attacks suggests that the similarity to normal web traffic presented particular challenges for distance-based algorithms. The uncertainty between these two web-based attack types highlights inherent traffic signature similarities that complicate ML classification.

Typically employing ping floods, DDoS_ICMP attacks overwhelm target systems through Internet Control Message Protocol (ICMP) packets. These attacks exhibited different traffic patterns, and most models achieved good detection accuracy. The slight confusion between DDoS_ICMP and DDoS_UDP observed in CNN and LSTM models (approximately 0.4% misclassification rate) suggests that minor temporal similarities can influence even advanced DL methods. DDoS_TCP attacks frequently employ Transmission Control Protocol (TCP) mechanisms to exhaust connection resources through SYN flooding attacks. Consistent misclassification patterns between DDoS_TCP and Port_Scanning attacks (ranging from 19.5% to 41.2% across KNN variants and SVM) reveal comparable network behavior traits. This confusion appears to result from both attack types involving systematic TCP port probing, though for distinct purposes while utilizing similar tech-

niques. DDoS_UDP attacks, which saturate target bandwidth through the User Datagram Protocol (UDP), displayed nearly flawless detection rates across all models in both base and enhanced configurations. Their uniquely identifiable traffic patterns, likely a result of UDP's connectionless nature, produce behaviors markedly different from other attack types. Feature space analysis provided important new insights into the identification of DDoS attack variants. For distinguishing DDoS attacks from normal traffic, network characteristics such as packet size distribution, inter-arrival time, protocol distribution, and flow duration demonstrated considerable discriminative potential. However, greater challenges arose in separating different DDoS variants from other attack categories.

Analysis of the principal components (PCA) of the feature space exposed some interesting trends in the distribution of attack types. DDoS_UDP attacks formed a distinct cluster, a separation that likely contributed to their consistently high detection rates across the tested models. DDoS_HTTP attacks, on the other hand, showed significant overlap with XSS traffic, particularly along components associated with HTTP header attributes and request patterns. Notably, this overlap persisted even after feature selection was applied, helping to explain the recurrent misclassification trends observed in the KNN variants. Furthermore, temporal features played a very important role in enhancing detection accuracy. Flow duration, packet rate, and burst patterns recorded the highest relevance scores within the RF model, emphasizing their value in distinguishing between legitimate and attack traffic. These sequential properties help to explain the remarkable early performance of LSTM networks, whose recurrent structure naturally captures temporal dependencies without requiring careful feature engineering. Figure 4 shows a comparison of training and detection times over all assessed models, therefore providing more understanding of the trade-offs between predictive performance and computational efficiency.

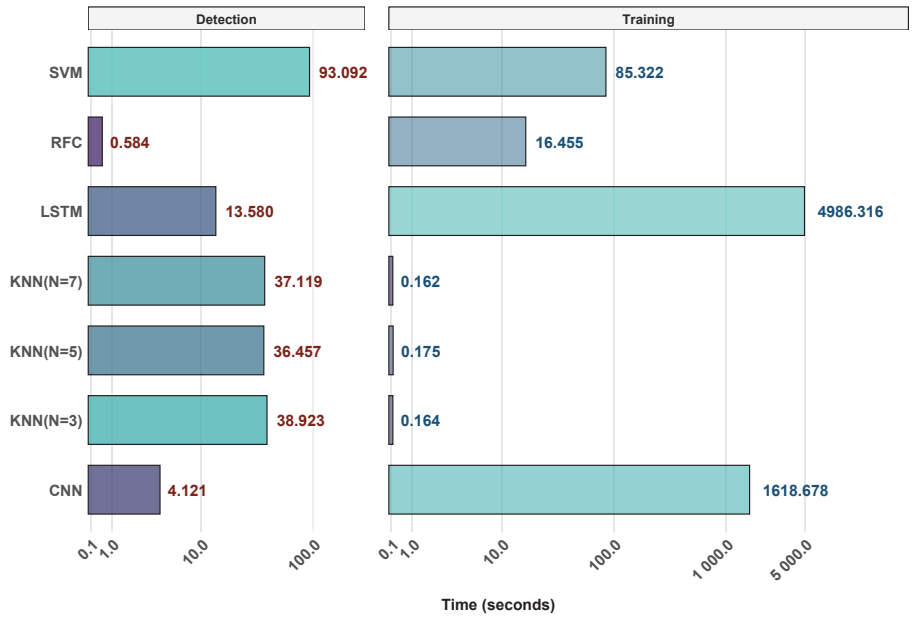


Figure 4. Comparison of processing times across different models.

6. Limitations and Future Work

Though the Edge-IIoTset dataset provides a rather structured view of attack paths, its reflection of the real-world diversity and complexity of DDoS threats targeting IoT environments remains limited. Particularly when considering adversarial strategies and concept drift, where attack behaviors evolve in unanticipated ways, controlled testing scenarios often fail to capture the adaptive nature of modern attacks. The models architec-

tures, which were precisely tuned to the dataset, represent another limitation. While they perform well within that controlled context, their generalizability remains questionable. Real-world IoT deployments often involve different traffic patterns, attack types, and noise levels compared to those represented in curated datasets. Without appropriate architectural adaptation, these differences can significantly impair model performance. Simply put, a model designed for one dataset is unlikely to hold up across different real-world scenarios.

Several key research directions for future exploration include the following: automated feature engineering using AutoML techniques, model compression and optimization strategies, hybrid model approaches, unified learning methodologies, and cross-dataset validation using diverse IoT security datasets. Cross-dataset evaluation, including validation on real-world IoT traffic from operational deployments, would enhance generalizability and provide insights into algorithmic performance across heterogeneous IoT environments. In particular, AutoML-driven feature engineering should be further investigated to help identify optimal feature subsets tailored to specific DDoS variants. Research should also be conducted on model compression and optimization techniques specifically tailored for DDoS detection in resource-constrained environments such as the case in IoT. It would also be beneficial to investigate ensemble approaches combining high-accuracy DL models with efficient traditional approaches. Further research on transfer learning techniques could help to achieve enhanced applicability across a range of IoT deployment scenarios by lowering the processing requirements of DL models while yet preserving detection performance.

7. Conclusions

DDoS attacks still form a critical challenge for network systems, especially IoT devices that usually exhibit insufficient protection mechanisms. This research evaluated the effectiveness of multiple ML and DL detection approaches. The RF model demonstrated exceptional performance, with a 99.99% accuracy rate and a remarkable detection time of 0.584 s, indicating that it is appropriate for real-time defense. Although the KNN model attained about 99.1% accuracy, its slower detection times (36 to 39 s) limit its applicability in situations requiring quick reactions. CNN achieved an accuracy of 99.80%, and LSTM achieved about 99.25%. However, both needed a lot of training time, more than 1600 s for CNN and about 4986 s for LSTM. Therefore, environments with strong computational resources are better suited for these DL approaches. The SVM showed a remarkable accuracy of 99.26%; however, its considerable training (85.32 s) and detection (93.09 s) duration make it less suited for fast detection uses. Because of the similarly structured network traffic patterns, all of the examined techniques encountered difficulties distinguishing between certain attack types, such as XSS attacks and DDoS_HTTP, and SQL injection and uploading attacks. The findings suggest that cybersecurity teams should carefully balance the needs for accuracy and processing power when selecting algorithms to defend against DDoS attacks. For most cases, RF remains the most sensible choice even if DL methods can be helpful in instances involving many resources and complex attacks.

Funding: This research was funded by the Public Authority for Applied Education and Training in Kuwait, grant number BS-23-01.

Data Availability Statement: The data presented in this study are openly available in Kaggle-Edge-IIoTset dataset, https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cybersecurity-dataset-of-iot-iiot, accessed on 12 March 2025.

Acknowledgments: The author acknowledges the support provided by the Public Authority for Applied Education and Training.

Conflicts of Interest: The author declares no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DDoS	Distributed Denial of Service
IoT	Internet of Things
IIoT	Industrial Internet of Things
ML	Machine Learning
DL	Deep Learning
RF	Random Forest
KNN	K-Nearest Neighbors
SVM	Support Vector Machine
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory
RNN	Recurrent Neural Network
DCNN/BiLSTM	Deep CNN with Bidirectional LSTM
LR	Logistic Regression
DT	Decision Tree
IDS	Intrusion Detection System
XSS	Cross-Site Scripting
SQL	Structured Query Language
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
AI	Artificial Intelligence
SMOTE	Synthetic Minority Oversampling Techniques
RNN	Recurrent Neural Networks
DNN	Deep Neural Networks
GDB	GradientBoosting
ADB	Adaboost
PCA	Principal Component Analysis

Appendix A. Class Distribution Data

Table A1. Class distribution in the Edge-IIoTset test set.

Attack Class	Number of Instances
Normal	4896
DDoS_UDP	2890
DDoS_ICMP	2554
DDoS_HTTP	2123
Uploading	2083
DDoS_TCP	2075
SQL_injection	2016
Password	1992
XSS	1979
Backdoor	1976
Vulnerability_scanner	1982
Ransomware	1934
Port_Scanning	1727
Fingerprinting	157
MITM	56
Total	**30,440**

Confusion Matrix Visualizations

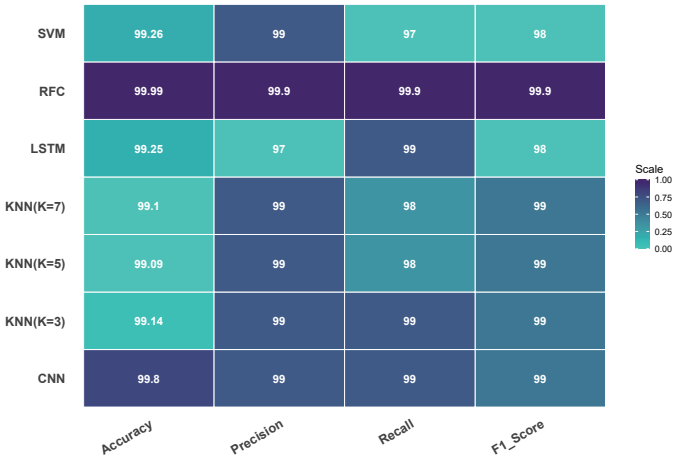


Figure A1. Heatmap of confusion matrix across all evaluated models. Note: Darker colors indicate better performance.



Figure A2. Heatmap of training and detection times across all evaluated models. Note: Darker colors indicate better performance.

Confusion Matrix for each model

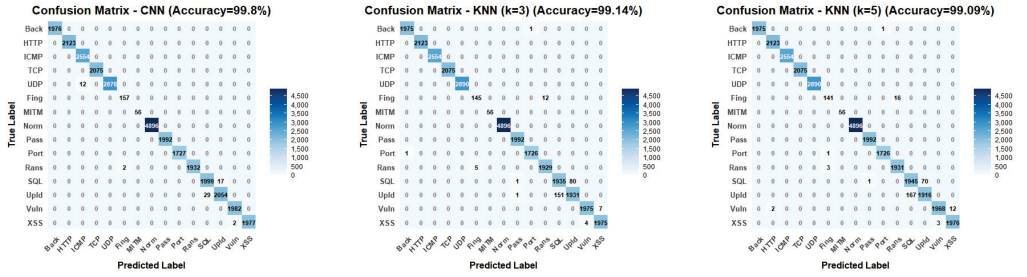


Figure A3. *Cont.*

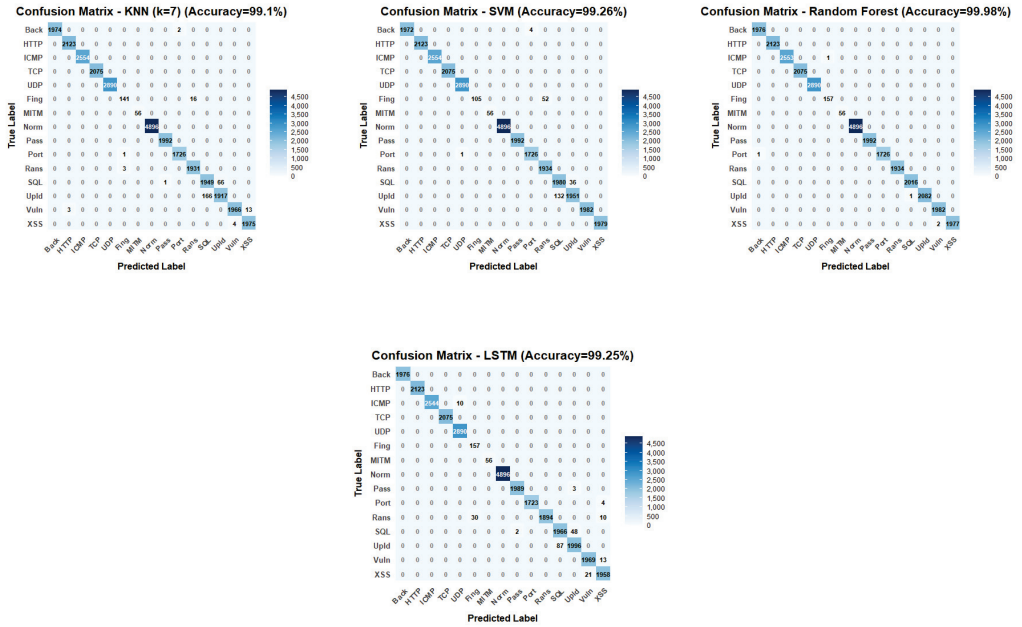


Figure A3. Confusion matrices comparison across all evaluated models: (top row) CNN, KNN (K = 3), KNN (K = 5); (middle row) KNN (K = 7), SVM, RF; (bottom) LSTM.

References

1. Adedeji, K.B.; Abu-Mahfouz, A.M.; Kurien, A.M. DDoS Attack and Detection Methods in Internet-Enabled Networks: Concept, Research Perspectives, and Challenges. *J. Sens. Actuator Netw.* **2023**, *12*, 51. [CrossRef]
2. Makhdoom, I.; Abolhasan, M.; Lipman, J.; Liu, R.P.; Ni, W. Anatomy of threats to the internet of things. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 1636–1675. [CrossRef]
3. Falowo, O.I.; Abdo, J.B. 2019–2023 in Review: Projecting DDoS Threats With ARIMA and ETS Forecasting Techniques. *IEEE Access* **2024**, *12*, 26759–26772. [CrossRef]
4. Sunny, B.; Krishan, K. Characterization and Comparison of DDoS Attack Tools and Traffic Generators: A Review. *Int. J. Netw. Secur.* **2017**, *19*, 383–393.
5. Peraković, D.; Periša, M.; Cvitić, I. Analysis of the IoT impact on volume of DDoS attacks. *XXXIII Simp. Novim Tehnol. Poštanskom i Telekomun. Saobraćaju–PosTel* **2015**, *2015*, 295–304.
6. Cloudflare. DDoS Threat Report for 2024 Q4. 2025. Available online: https://blog.cloudflare.com/ddos-threat-report-for-2024-q4/ (accessed on 12 March 2025).
7. Hafsa, M.; Malik, S. Factors Effecting Businesses due to Distributed Denial of Service (DDoS) Attack. In Proceedings of the International Conference on Intelligent Computing (ICIC), Lahore, Pakistan, 9–10 November 2021. [CrossRef]
8. Somani, G.; Gaur, M.S.; Sanghi, D. DDoS/EDoS attack in cloud: Affecting everyone out there! In Proceedings of the 8th International Conference on Security of Information and Networks, SIN '15, Sochi, Russia, 8–10 September 2015. [CrossRef]
9. Zhao, K.; Ge, L. A Survey on the Internet of Things Security. In Proceedings of the 2013 Ninth International Conference on Computational Intelligence and Security, Emeishan, China, 14–15 December 2013; pp. 663–667. [CrossRef]
10. Hassija, V.; Chamola, V.; Saxena, V.; Jain, D.; Goyal, P.; Sikdar, B. A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. *IEEE Access* **2019**, *7*, 82721–82743. [CrossRef]
11. Gartner. Forecast: Internet of Things endpoints and spending, worldwide, 2020. *Gart. Forecast. Anal.* **2020**, *11*, 1–13.
12. Djenna, A.; Harous, S.; Saidouni, D.E. Internet of things meet internet of threats: New concern cyber security issues of critical cyber infrastructure. *Appl. Sci.* **2021**, *11*, 4580. [CrossRef]
13. Bassene, A.; Gueye, B. DeepDDoS: A Deep-Learning Model for Detecting Software Defined Healthcare IoT Networks Attacks. In *Advances in Ubiquitous Networking*; Springer: Cham, Switzerland, 2021. [CrossRef]
14. Razavi, H.; Jamali, M.R.; Emsaki, M.; Ahmadi, A.; Hajiaghei-Keshteli, M. Quantifying the Financial Impact of Cyber Security Attacks on Banks: A Big Data Analytics Approach. In Proceedings of the 2023 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Regina, SK, Canada, 24–27 September 2023; pp. 533–538.
15. Najafimehr, M.; Zarifzadeh, S.; Mostafavi, S. DDoS attacks and machine learning based detection methods: A survey and taxonomy. *Eng. Rep.* **2023**, *5*, e12697. [CrossRef]

16. Karie, N.M.; Sahri, N.M.; Haskell-Dowland, P. IoT Threat Detection Advances, Challenges and Future Directions. In Proceedings of the 2020 Workshop on Emerging Technologies for Security in IoT (ETSecIoT), Sydney, NSW, Australia, 21 April 2020.
17. Asharf, J.; Moustafa, N.; Khurshid, H.; Debie, E.; Haider, W.; Wahab, A. A Review of Intrusion Detection Systems Using Machine and Deep Learning in Internet of Things: Challenges, Solutions and Future Directions. *Electronics* **2020**, *9*, 1177. [CrossRef]
18. Samtani, S.; Kantarcioglu, M.; Chen, H. Trailblazing the artificial intelligence for cybersecurity discipline: A multi-disciplinary research roadmap. *ACM Trans. Manag. Inf. Syst.* **2020**, *11*, 1–19. [CrossRef]
19. Rasmussen, C.B.; Moeslund, T.B. Evaluation of Model Selection for Kernel Fragment Recognition in Corn Silage. *arXiv* **2020**, arXiv:2004.00292.
20. Mool, A.; Panda, J.; Sharma, K. Face Detection framework for accelerated analysis of High quality Multimedia content. *Int. J. Pattern Recognit. Artif. Intell.* **2024**, *38*, 2456001. [CrossRef]
21. Faiz, M.N.; Somantri, O.; Supriyono, A.R.; Muhammad, A.W. Impact of feature selection methods on machine learning-based for detecting DDoS attacks: Literature review. *J. Inform. Telecommun. Eng.* **2022**, *5*, 305–314. [CrossRef]
22. Ferrag, M.A.; Friha, O.; Hamouda, D.; Maglaras, L.; Janicke, H. Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. *IEEE Access* **2022**, *10*, 40281–40306. [CrossRef]
23. Zarpelão, B.B.; Miani, R.S.; Kawakani, C.T.; de Alvarenga, S.C. A survey of intrusion detection in Internet of Things. *J. Netw. Comput. Appl.* **2017**, *84*, 25–37. [CrossRef]
24. Bhunia, S.S.; Gurusamy, M. Dynamic attack detection and mitigation in IoT using SDN. In Proceedings of the 2019 27th International Telecommunication Networks and Applications Conference (ITNAC), Melbourne, VIC, Australia, 22–24 November 2017; pp. 1–6.
25. Sagirlar, G.; Carminati, B.; Ferrari, E. AutoBotCatcher: Blockchain-based P2P botnet detection for the Internet of Things. In Proceedings of the 2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC), Philadelphia, PA, USA, 18–20 October 2018; pp. 1–8.
26. Al-Garadi, M.A.; Mohamed, A.; Al-Ali, A.K.; Du, X.; Ali, I.; Guizani, M. A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 1646–1685. [CrossRef]
27. Adat, V.; Gupta, B.B. Security in Internet of Things: Issues, challenges, taxonomy, and architecture. *Telecommun. Syst.* **2018**, *67*, 423–441. [CrossRef]
28. Kumar, P.; Tripathi, R.; Singh, G. A distributed framework for detecting DDoS attacks in smart home IoT devices. *J. Supercomput.* **2019**, *76*, 4784–4810.
29. Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J. Survey of intrusion detection systems: Techniques, datasets and challenges. *Cybersecurity* **2019**, *2*, 20. [CrossRef]
30. Day, D.; Flores, D. Tactics and signatures: Defending against DDoS with effective mitigations. *J. Cybersecur. Res.* **2017**, *2*, 12–24.
31. Hajiheidari, S.; Wakil, K.; Badri, M.; Navimipour, N.J. Intrusion detection systems in the Internet of things: A comprehensive investigation. *Comput. Netw.* **2019**, *160*, 165–191. [CrossRef]
32. Koroniotis, N.; Moustafa, N.; Sitnikova, E.; Turnbull, B. Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Gener. Comput. Syst.* **2019**, *100*, 779–796. [CrossRef]
33. Kim, J.; Kim, J.; Kim, H.; Shim, M.; Choi, E. CNN-based network intrusion detection against denial-of-service attacks. *Electronics* **2020**, *9*, 916. [CrossRef]
34. Yin, C.; Zhu, Y.; Fei, J.; He, X. A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access* **2018**, *5*, 21954–21961. [CrossRef]
35. Ferrag, M.A.; Maglaras, L.; Moschoyiannis, S.; Janicke, H. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *J. Inf. Secur. Appl.* **2020**, *50*, 102419. [CrossRef]
36. Latif, S.; Zou, Z.; Idrees, Z.; Ahmad, J. A novel attack detection scheme for the industrial internet of things using a lightweight random neural network. *IEEE Access* **2020**, *8*, 89337–89350. [CrossRef]
37. Moustafa, N.; Turnbull, B.; Choo, K.K.R. An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things. *IEEE Internet Things J.* **2019**, *6*, 4815–4830. [CrossRef]
38. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
39. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
40. Ghosh, S.; Dasgupta, A.; Swetapadma, A. A Study on Support Vector Machine based Linear and Non-Linear Pattern Classification. In Proceedings of the 2019 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 21–22 February 2019; pp. 24–28. [CrossRef]
41. Abu Alfeilat, H.A.; Hassanat, A.B.; Lasassmeh, O.; Tarawneh, A.S.; Alhasanat, M.B.; Eyal Salman, H.S.; Prasath, V.S. Effects of distance measure choice on k-nearest neighbor classifier performance: A review. *Big Data* **2019**, *7*, 221–248. [CrossRef]
42. Mucherino, A.; Papajorgji, P.J.; Pardalos, P.M.; Mucherino, A.; Papajorgji, P.J.; Pardalos, P.M. K-nearest neighbor classification. In *Data Mining in Agriculture*; Springer: New York, NY, USA, 2009; pp. 83–106.
43. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]

44. Liaw, A.; Wiener, M. Classification and regression by randomForest. *R News* **2002**, *2*, 18–22.
45. Biau, G.; Scornet, E. A random forest guided tour. *Test* **2016**, *25*, 197–227. [CrossRef]
46. Rigatti, S.J. Random forest. *J. Insur. Med.* **2017**, *47*, 31–39. [CrossRef]
47. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1989**, *1*, 541–551. [CrossRef]
48. Chauhan, N.K.; Singh, K. A review on conventional machine learning vs deep learning. In Proceedings of the 2018 International Conference on Computing, Power and Communication Technologies (GUCON), Greater Noida, India, 28–29 September 2018; pp. 347–352.
49. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. [CrossRef]
50. Xenofontos, C.; Zografopoulos, I.; Konstantinou, C.; Jolfaei, A.; Khan, M.K.; Choo, K.K.R. Consumer, Commercial, and Industrial IoT (In)Security: Attack Taxonomy and Case Studies. *IEEE Internet Things J.* **2021**, *9*, 199–221. [CrossRef]
51. Vishwakarma, R.; Jain, A.K. A survey of DDoS attacking techniques and defence mechanisms in the IoT network. *Telecommun. Syst.* **2020**, *73*, 3–25. [CrossRef]
52. Obaidat, M.A.; Obeidat, S.; Holst, J.; Al Hayajneh, A.; Brown, J. A Comprehensive and Systematic Survey on the Internet of Things: Security and Privacy Challenges, Security Frameworks, Enabling Technologies, Threats, Vulnerabilities and Countermeasures. *Computers* **2020**, *9*, 44. [CrossRef]
53. Abdul-Ghani, H.A.; Konstantas, D.; Mahyoub, M. A comprehensive IoT attacks survey based on a building-blocked reference model. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 355–373.
54. Khanam, S.; Ahmedy, I.B.; Idris, M.Y.I.; Jaward, M.H.; Sabri, A.Q.B.M. A Survey of Security Challenges, Attacks Taxonomy and Advanced Countermeasures in the Internet of Things. *IEEE Access* **2020**, *8*, 219709–219743. [CrossRef]
55. Das, A.K.; Zeadally, S.; He, D. Taxonomy and analysis of security protocols for Internet of Things. *Future Gener. Comput. Syst.* **2018**, *89*, 110–125. [CrossRef]
56. Bovenzi, G.; Aceto, G.; Ciuonzo, D.; Montieri, A.; Persico, V.; Pescapé, A. Network anomaly detection methods in IoT environments via deep learning: A fair comparison of performance and robustness. *Comput. Secur.* **2023**, *128*, 103167. [CrossRef]
57. Nawir, M.; Amir, A.; Yaakob, N.; Lynn, O.B. Internet of Things (IoT): Taxonomy of security attacks. In Proceedings of the 2016 3rd International Conference on Electronic Design (ICED), Phuket, Thailand, 11–12 August 2016; pp. 321–326.
58. Sasi, T.; Lashkari, A.H.; Lu, R.; Xiong, P.; Iqbal, S. A comprehensive survey on IoT attacks: Taxonomy, detection mechanisms and challenges. *J. Inf. Intell.* **2024**, *2*, 455–513. [CrossRef]
59. Prajapati, P.M.; Gandhi, P.P.; Degadwala, S. Exploring Methods of Mitigation against DDoS Attack in an IoT Network. In Proceedings of the 2024 International Conference on Inventive Computation Technologies (ICICT), Lalitpur, Nepal, 24–26 April 2024; pp. 1373–1377. [CrossRef]
60. Giachoudis, N.; Damiris, G.P.; Theodoridis, G.; Spathoulas, G. Collaborative Agent-based Detection of DDoS IoT Botnets. In Proceedings of the 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini, Greece, 29–31 May 2019; pp. 205–211. [CrossRef]
61. González-Landero, F.; García-Magariño, I.; Lacuesta, R.; Lloret, J. ABSDDoS: An AgentBased Simulator about Strategies of Both DDoS Attacks and Their Defenses, to Achieve Efficient Data Forwarding in Sensor Networks and IoT Devices. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 7264269. [CrossRef]
62. Sonar, K.; Upadhyay, H. An Approach to Secure Internet of Things Against DDoS. In *Advances in Intelligent Systems and Computing*; Springer: Singapore, 2016; pp. 367–376. [CrossRef]
63. Spathoulas, G.; Giachoudis, N.; Damiris, G.P.; Theodoridis, G. Collaborative Blockchain-Based Detection of Distributed Denial of Service Attacks Based on Internet of Things Botnets. *Future Internet* **2019**, *11*, 226. [CrossRef]
64. Doshi, K.; Yilmaz, Y.; Uludag, S. Timely Detection and Mitigation of Stealthy DDoS Attacks via IoT Networks. *IEEE Trans. Dependable Secur. Comput.* **2021**, *18*, 2164–2176. [CrossRef]
65. Zaidi, S.A.N. Mitigating DDoS Attacks on IoT Networks: Strategies and Solutions. *Int. J. Res. Appl. Sci. Eng. Technol.* **2024**, *12*, 1387–1394. [CrossRef]
66. Rashid, M.M.; Khan, S.U.; Eusufzai, F.; Redwan, M.A.; Sabuj, S.R.; Elsharief, M. A Federated Learning-Based Approach for Improving Intrusion Detection in Industrial Internet of Things Networks. *Network* **2023**, *3*, 158–179. [CrossRef]
67. Tareq, I.; Elbagoury, B.M.; El-Regaily, S.; El-Horbaty, E.S.M. Analysis of ton-iot, unw-nb15, and edge-iiot datasets using dl in cybersecurity for iot. *Appl. Sci.* **2022**, *12*, 9572. [CrossRef]
68. Hnamte, V.; Hussain, J. DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system. *Telemat. Inform. Rep.* **2023**, *10*, 100053. [CrossRef]

69. Khacha, A.; Saadouni, R.; Harbi, Y.; Aliouat, Z. Hybrid deep learning-based intrusion detection system for industrial internet of things. In Proceedings of the 2022 5th International Symposium on Informatics and its Applications (ISIA), M'sila, Algeria, 29–30 November 2022; pp. 1–6.
70. Thiyam, B.; Dey, S. Efficient feature evaluation approach for a class-imbalanced dataset using machine learning. *Procedia Comput. Sci.* **2023**, *218*, 2520–2532. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Article

Intrusion Detection in IoT Using Deep Residual Networks with Attention Mechanisms

Bo Cui [1,2], Yachao Chai [1,2], Zhen Yang [1,2] and Keqin Li [3,*]

[1] College of Computer Science, Inner Mongolia University, Hohhot 010021, China; cscb@imu.edu.cn (B.C.); 32009072@mail.imu.edu.cn (Y.C.); 32209059@mail.imu.edu.cn (Z.Y.)
[2] Engineering Research Center of Ecological Big Data, Ministry of Education, Hohhot 010021, China
[3] Department of Computer Science, State University of New York, New Paltz, NY 12561, USA
* Correspondence: lik@newpaltz.edu

Abstract: Connected devices in IoT systems usually have low computing and storage capacity and lack uniform standards and protocols, making them easy targets for cyberattacks. Implementing security measures like cryptographic authentication, access control, and firewalls for IoT devices is insufficient to fully address the inherent vulnerabilities and potential cyberattacks within the IoT environment. To improve the defensive capabilities of IoT systems, some research has focused on using deep learning techniques to provide new solutions for intrusion detection systems. However, some existing deep learning-based intrusion detection methods suffer from inadequate feature extraction and insufficient model generalization capability. To address the shortcomings of existing detection methods, we propose an intrusion detection model based on temporal convolutional residual modules. An attention mechanism is introduced to assess feature scores and enhance the model's ability to concentrate on critical features, thereby boosting its detection performance. We conducted extensive experiments on the ToN_IoT dataset and the UNSW-NB15 dataset, and the proposed model achieves accuracies of 99.55% and 89.23% on the ToN_IoT and UNSW-NB15 datasets, respectively, with improvements of 0.14% and 15.3% compared with the current state-of-the-art models. These results demonstrate the superior detection performance of the proposed model.

Keywords: IoT; cyber attacks; intrusion detection; deep learning; attention mechanism

1. Introduction

With the leap forward in next-generation information technology such as 5G, artificial intelligence, and big data, IoT has entered a new era of development. However, connected devices in IoT systems often have low computing and storage capacity and lack uniform standards and protocols, making them easy targets for cyber attacks. Secondly, IoT device manufacturers do not implement strict security measures when producing devices, which makes current IoT devices have many vulnerabilities, which also leads to many potential security threats in the IoT system composed of IoT devices [1]. For example, the Mirai botnet is capable of executing large-scale distributed denial of service attacks via IoT devices [2]; IoT devices are usually connected through wireless networks, and intruders can obtain private information from communication channels through eavesdropping [3]. These cyberattacks not only disrupt the availability of device functionality but may also steal important information and data about the device and the users who use it. Therefore, designing a security approach that can effectively address the cyberattacks faced by IoT has been the focus of research [4].

Implementing security measures such as cryptographic authentication, access control, and firewalls for IoT devices does not fully address the cyberattack problem or guarantee absolute security in the IoT environment [5]. To improve the defense capability of IoT systems, many studies have focused on machine learning-based or deep learning-based intrusion detection methods [6]. An intrusion detection system (IDS) [7] is a network

security device that continuously monitors network traffic and issues alerts or takes proactive measures when suspicious activities are detected. The difference with other network security measures is that IDS is a proactive security protection technology. IDS solutions can be classified into three approaches: signature-based, anomaly-based, and hybrid [6]. In general, signature-based approaches are effective against known attacks. However, due to the heterogeneity, dynamics, and complexity of IoT networks, the signature-based approach is less efficient for IoT because it requires continuous human intervention to extract attack patterns and update the IDS model. On the other hand, anomaly-based approaches are effective for unknown attacks, making them advantageous in IoT as they can detect zero-day attacks with minimal human intervention. The hybrid approach combines signature-based and anomaly-based techniques. However, the use of signature-based intrusion detection methods in IoT environments is limited due to their inability to detect unknown attacks. Therefore, anomaly-based intrusion detection systems play a crucial role in IoT security.

Some existing anomaly intrusion detection systems use traditional machine learning techniques to build IDS models [8]. These traditional techniques, however, encounter challenges with the high speed and volume of data generated by IoT devices. They rely extensively on feature engineering to extract representative features from the unstructured data, leading to performance degradation when applied to large-scale and high-dimensional datasets. Consequently, several studies have shifted focus toward employing deep learning techniques to develop more effective solutions [9].

Deep learning automates feature extraction and classification by designing multilayer neural networks into architectures and using large amounts of data and computation. The advantage of this approach is that it can perform feature learning without human intervention and can achieve good results when dealing with large or complex datasets. The powerful data processing and feature learning capabilities and the ability to detect unknown attacks that deep learning has can provide advanced solutions for IoT intrusion detection [9–12]. Therefore, researchers have applied deep learning to the field of IoT intrusion detection and have achieved good results [9–14].

However, there are several obvious problems with some current deep learning-based IoT intrusion detection methods. First, most of the current methods use previous data sources, i.e., NSL-KDD, KDD-99, etc., for evaluation [9–13], which do not contain current and up-to-date attack data against the IoT. Secondly, the detection models mentioned in a significant number of methods have a complex structure [11–14], which leads to the limitation of the application of the models in the IoT environment. In addition, some methods ignore the fact that IoT traffic data has both spatial and temporal features [12], which leads to inadequate feature extraction by the models proposed in the methods and thus affects the detection performance. Moreover, some models cannot be processed for heterogeneous data in the IoT environment, which leads to a slight lack of model generalization [15], i.e., the models are not sufficiently adaptable to new data and have poor detection performance on new datasets. Finally, the limited amount of labeled data available for training deep learning models in the IoT environment seriously affects the detection performance of the classifier. In our work, we proposed an improved residual network structure, which consists of three residual modules, each containing a CONV-LSTM sub-network, and connects the module design to the attention module, which greatly reduces the complexity of the model structure. The evaluation results based on ToN_IoT with the UNSW-NB15 dataset show that our model has better performance performance than existing models. The contributions of our work are as follows:

- Considering the spatiotemporal characteristics of IoT traffic data, we proposed an improved residual network structure that avoids the performance impact of extracting only a single feature.
- We introduced an attention mechanism in the model to compute weights representing the importance of different features to help the model focus on the most important features.

- Higher detection accuracy. The performance of the algorithm proposed in this paper, in terms of detection accuracy, is superior to some current state-of-the-art methods.
- Stronger generalization ability. With a certain amount of data, this paper improves the model's expressiveness by increasing the network width and optimizing the loss function to reach the global optimum.

The rest of this paper is organized as follows. Section 2 describes the intrusion detection-related work. Section 3 describes the mathematical modeling of the proposed model. Section 4 presents the experimental results and discussion. Finally, Section 5 summarizes the contributions and results of this study.

2. Related Work

In recent years, Hasan et al. [16] studied attack detection models in IoT sensors using machine learning methods. They accurately compared the performance of several machine learning techniques in predicting attacks and violations in IoT systems. A robust algorithm was developed for detecting IoT cyberattacks, particularly focusing on virtual environments. The proposed system demonstrated better detection accuracy compared to existing models. Ravi et al. [17] proposed DDoS attack mitigation and learning-driven detection in IoT through SDN-Cloud architecture. This approach aims at detecting DDoS attacks launched in IoT servers using malicious wireless IoT. Zhang et al. [18] proposed an effective method for classifying network traffic, which uses principal components analysis (PCA) to remove irrelevant features and Gaussian Parsimonious Bayes as a classifier. However, machine learning methods have certain limitations. Firstly, their performance is heavily dependent on the robustness of the feature engineering techniques employed. Secondly, their effectiveness diminishes when applied to large-scale and high-dimensional data. Lastly, their learning capability is insufficient to effectively handle unknown attacks in the IoT environment.

The most important advantage of deep learning over traditional machine learning is its superior performance on large datasets. The use of IoT systems usually generates a large amount of data, which is more complex and diverse, and contains a variety of intrusive behaviors. The ability of deep learning to automatically model complex feature sets from sample data makes deep learning more relevant in IoT security applications. For example, Mohamed et al. [19] proposed the use of DeepIFS integrated with gated recurrent units and a multi-headed attention mechanism to detect intrusions. Liu et al. [20] introduced a federated learning approach for collaborative and decentralized training on edge devices. They utilized LSTM to capture temporal representations and employed attention-enhanced convolutional neural networks to learn important spatial information. The study of Recurrent Neural Networks and their variants is important for improving the security of IoT systems, especially against time-series-based threats. For example, Yan et al. [21] used a variational autoencoder (VAE) as a network baseline and then learned potential temporal representations of the input time series for intrusion detection by RNNs. They also used FNN to parameterize the mean and variance of each time window to provide a non-smooth architecture that can operate in the absence of persistent noise. Gao et al. [22] introduced the LSTM-GaussianNB architecture for evaluating the probability of outliers in IoT data. These research works further demonstrate the suitability of recurrent neural networks as an IoT intrusion detection model. Therefore, some researchers have gone a step further and combined recurrent neural networks with other methods to achieve better detection in the field of IoT intrusion detection. For example, Parra et al. [23] proposed a distributed cloud-based approach based on CNN and LSTM to detect and mitigate phishing and botnet attacks on client devices, which outperformed a single LSTM model.

In the last two years of research work, Khan et al. [24] proposed an efficient model called XSRU-IoMT for the effective and timely detection of complex attack traffic in medical IoT. However, the authors tested it using only a single dataset, which was not sufficient to demonstrate the effectiveness of the model. Wu et al. [25] proposed a hierarchical CNN-RNN neural network LuNet, but the generalization ability of the model was poor. Later,

Wu et al. proposed a dense residual network Densely-ResNet [15] based on LuNet for secure detection of edge, cloud, and fog layers. Although the generalization ability of the model was improved, it was slightly inadequate and the structure of the model was too complex. Latif et al. [26] pointed out that their proposed dense random neural network (DnRaNN) could improve the generalization performance of the model, but there were not enough experiments to show that the generalization ability of the model was improved. Therefore, to further improve the detection accuracy of the model while maintaining good generalization ability, this paper proposes an intrusion detection model based on temporal convolutional residual modules, which consists of three residual modules, each containing a CONV-LSTM subnetwork, and connects the module design to the attention module, which greatly reduces the complexity of the model structure. The model can effectively learn the spatiotemporal representation of IoT data and obtain high accuracy while maintaining good generalization performance. All experiments are conducted on the ToN_IoT dataset and the UNSW-NB15 dataset.

3. Proposed Model

Model Overview

To achieve intrusion detection in the IoT environment, researchers have introduced convolutional neural networks [27]. However, CNNs are usually used for feature extraction in static environments and lack long-term relevance storage mechanisms, which are not suitable for modeling sequential data in IoT. Some researchers have applied RNNs to the field of IoT intrusion detection for the sequential relationship of traffic data in the IoT environment. RNNs integrate a temporal layer to capture sequential data, learning multifaceted changes through the hidden units of recurrent units. These hidden units are modified based on the data provided to the network and are continuously updated. RNNs are used for IoT security due to their efficient management of sequential data. The study of RNNs and their variants is important to improve the security of IoT systems, especially against time-series-based threats [14]. However, RNNs suffer from the gradient disappearance or explosion problem, where gradients can become too small or too large during training, leading to their unsatisfactory predictions for IoT intrusion detection.

This paper studies intrusion detection models based on the combination of deep learning and attention mechanisms [28] for intrusion detection from IoT traffic data. For the characteristics of data in IoT, this paper combines CNN and LSTM units, redesigns and improves the ResNet architecture [29], and introduces an attention mechanism for the extracted high-dimensional features to give weight information to different features to avoid the problem of failing to express important information due to the excessive dimensionality of the features.

The ResNet model proposed in this paper consists of three main pairs of modules. Each module pair contains a residual module and a ResBlock-CBAM module. A jump connection is used between each two module pairs, and after the summation operation is performed on the output of the last module pair, the output of the module pair is classified by the classification layer, which finally constitutes the main structure of the deep residual network model used in this paper. The overall architecture of the model is shown in Figure 1. This network model overcomes the disadvantages of traditional deep learning, such as gradient disappearance and gradient explosion, and has a strong generalization capability.

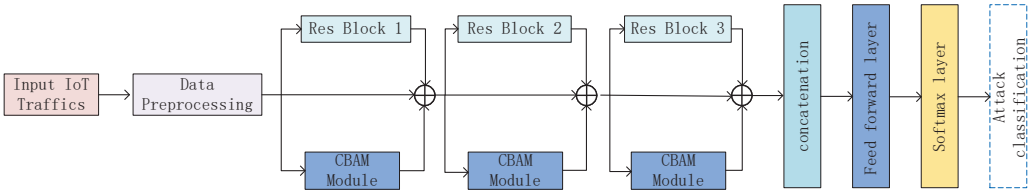


Figure 1. Model architecture diagram.

4. Residual Network Architecture

The specific structure design of the residual module proposed in this paper is shown in Figure 2. Each residual module consists of a CNN unit and an LSTM unit. The CNN unit mainly consists of three convolutional layers: the first convolutional layer has a convolutional kernel size of 1 × 1, which is used to reduce the dimensionality of the input feature map; the second convolutional layer has a convolutional kernel size of 3 × 3, which is the core part of the CNN structure and is used for feature extraction and nonlinear transformation; the third convolutional layer has a convolutional kernel size of 1 × 1, which is used to restore the dimensionality of the feature map to its original size. The proposed model extracts the features of the input data through the convolutional and pooling layers, and then the feature map is transformed and input to the LSTM unit.

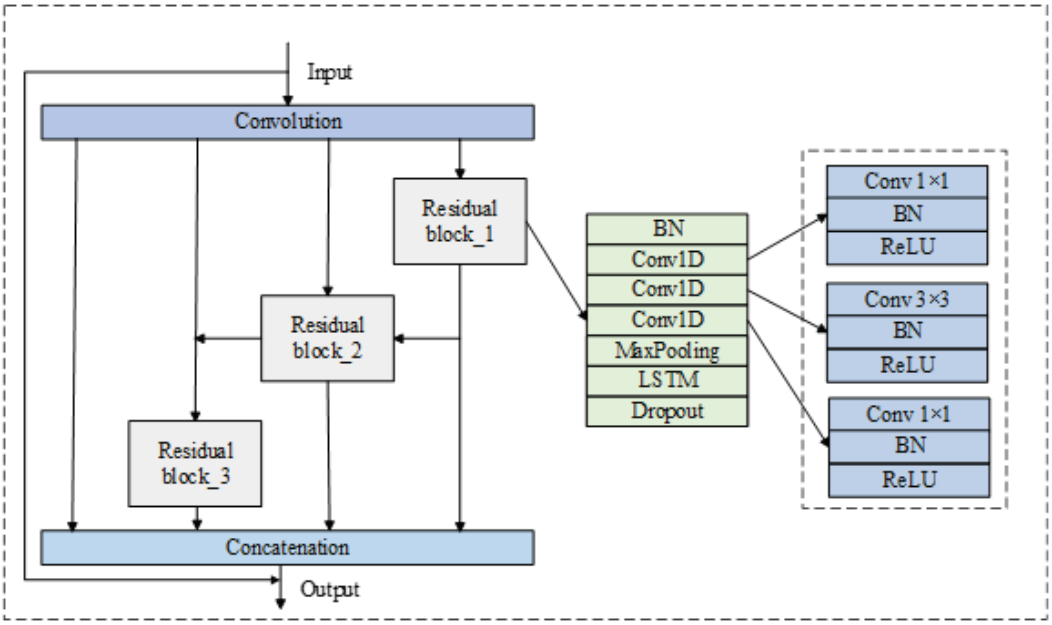


Figure 2. Residual network structure diagram.

Assuming that the input datum is x, the convolution layer in the proposed model extracts the spatial features of the given data and performs the convolution operation on the input data to obtain the output:

$$X = f(w_i \otimes x_i + b_i), \tag{1}$$

where $f(\cdot)$ is the activation function, w_i is the convolution kernel weight, and b_i is the corresponding bias. The rectified linear unit (ReLU) is used as the activation function:

$$f(x) = \max(0, X). \tag{2}$$

The first convolution operation is performed in the residual unit, using a 1 × 1 convolution kernel, reducing the number of channels of the input feature map to C/4 and the size of the output feature map to H × W × C/4; batch normalization (BN) and ReLU activation operations are performed on the output feature map:

$$y_i = \lambda x_i' + \varphi, \tag{3}$$

where λ and φ are learning parameters and x_i' is calculated as follows:

$$\mu_\beta = \frac{1}{m}\sum_{i=1}^{m} x_i. \tag{4}$$

$$\sigma_\beta^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_\beta)^2. \tag{5}$$

$$x_i' = \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 - \varepsilon}}. \tag{6}$$

In these equations, μ_β represents the mean of the mini-batch, which is the average value of the β-th feature map element in a mini-batch; σ_β^2 represents the variance of the mini-batch, which is the variance of the β-th feature map element in a mini-batch; x_i represents the i-th element of the input feature map; x_i' represents the i-th element of the normalized input feature map; m represents the size of the mini-batch in batch normalization; that is, the number of elements in the input feature map; and ε represents a small constant in batch normalization used to prevent division by zero.

The second and third convolution operations are performed using 3 × 3 and 1 × 1 convolution kernels, respectively, and each convolution operation is followed by a BN and ReLU activation operation. The output of the previous layer is then downsampled by the pooling layer to compress its features, making the data feature dimensionality reduced by:

$$h_j^l = down\left(h_j^{(l-1)} M^l\right), \tag{7}$$

where M^l denotes the size of the l pooling layer and $down(\cdot)$ denotes the downsampling function.

The maximum pooling layer is a common down-sampling technique in convolutional neural networks, which serves to reduce the feature dimension and improve the computational efficiency and generalization ability of the model. The specific roles of the maximum pooling layer are as follows: feature compression: the maximum pooling layer can reduce the size of the feature map to reduce the computation of the model, and it can retain the main features of the original data to avoid overfitting. Feature invariance: The maximum pooling layer can improve the feature invariance, i.e., the features can remain unchanged when the input changes slightly, thus improving the model's robustness and generalization ability. Feature selection: The maximum pooling layer can select the most important features in the data, i.e., keep the maximum value and ignore other values, thus improving the expressiveness and performance of the model. In summary, the maximum pooling layer can optimize the performance and efficiency of the convolutional neural network by compressing the feature map, improving the feature invariance, and selecting the most important features. The specific computational procedure is as follows:

The pooling operation converts M to $Z = [z^1, z^2, z^3, \ldots, z^C]$ of size $1 \times 1 \times C$, where z is calculated as follows:

$$z^C = \max(m_{i,j}). \tag{8}$$

After local features are extracted by the CNN, long-distance dependencies of these local features are captured using the LSTM. The following outlines the process of transforming the specific input features of the LSTM unit:

The input vector X at the current moment is input to the LSTM cell. The output value i_t of the input gate of the LSTM cell is calculated by the sigmoid function, and this value determines how the input vector x affects the state C_t. i_t ranges between 0 and 1. The output value f_t of the forgetting gate is computed by the sigmoid function. This value determines which information in the state C_{t-1} of the previous moment needs to be forgotten. f_t ranges from 0 to 1.

Updating the state. The current moment's state C_t can be updated by adding the result of the dot product of i_t and X to the previous moment's state C_{t-1} and subtracting the result of the dot product of f_t and the previous moment's state C_{t-1}. The output value o_t of the output gate is calculated by the sigmoid function, which determines which information

in the current moment's state C_t needs to be output. o_t ranges from 0 to 1. The hidden state h_t of the current moment can be obtained by performing a dot product operation between the state C_t of the current moment and the output value o_t of the output gate. Finally, the output is passed to the next layer for calculation.

In this way, the input features can be better represented by the transformation of the LSTM cells, thus improving the performance of the model.

The dropout layer is used to avoid the overfitting problem during training and to improve the model generalization ability. ReLU is then chosen as the activation function to overcome the gradient disappearance problem and to speed up the training speed.

5. Convolutional Block Attention Mechanism

In the IoT environment, network traffic data often contains different features that are of different importance for intrusion detection. Therefore, this section introduces the CBAM module [30] in the model to help the model distinguish the importance of features in order to obtain important features and improve the performance of the model. As shown in Figure 3. The structures of the channel attention module and the spatial attention module are shown in Figures 4 and 5, respectively. The specific computational procedure of the CBAM module is described as follows: The specific structure of the channel attention module is shown in Figure 4. For a given feature map F, the channel attention module first performs Global Average Pooling (GAP) to obtain the average value of each channel, and then performs feature mapping through two fully connected layers to obtain the weight coefficients of each channel. Finally, the weight coefficients are applied to the feature map to emphasize the important channels and suppress the unimportant ones. The whole process is as follows:

$$F' = M_C(F) \otimes F. \tag{9}$$

$$F'' = M_C(F') \otimes F'. \tag{10}$$

where $\otimes$ denotes element-by-element multiplication. The channel attention module focuses on what is meaningful in the input data. CBAM has two pools: maximum pooling (MaxPool) and average pooling (AvgPool). Pooling allows extracting high-level features, and different pooling means extracting richer high-level features.

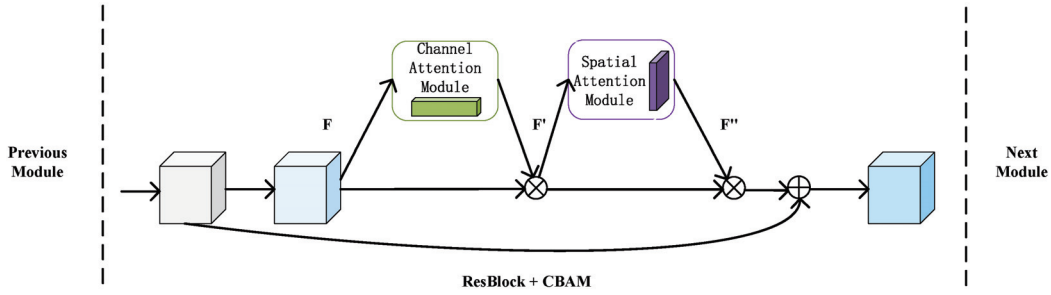


Figure 3. CBAM module diagram.

Since each channel of the feature map extracts some level of feature information, the channel attention mechanism feature information is more important. In order to compute channel attention efficiently, the channel attention module uses a spatial dimensionality method that compresses the input feature mapping, compared to using a single pooling method the channel attention module uses average pooling and maximum pooling methods and proves that the dual pooling method has stronger representational power. The specific computational procedure is shown as follows:

$$M_C(F) = \sigma(\mathrm{MLP}(\mathrm{AvgPool}(F)) + \mathrm{MLP}(\mathrm{MaxPool}(F))). \tag{11}$$

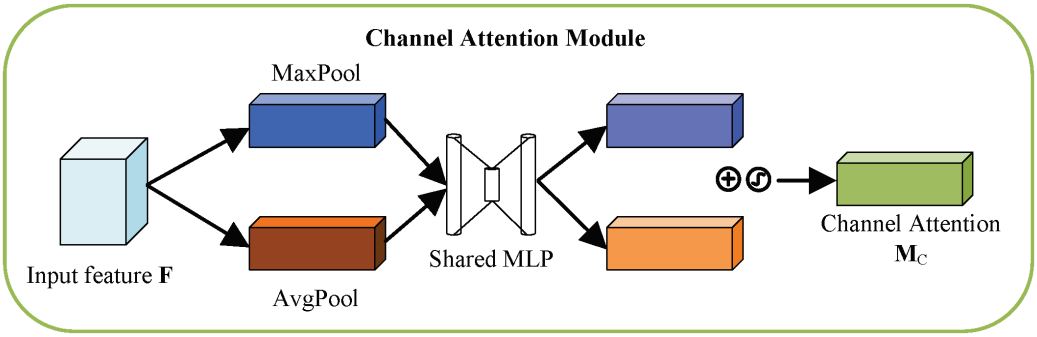


Figure 4. Channel attention module diagram.

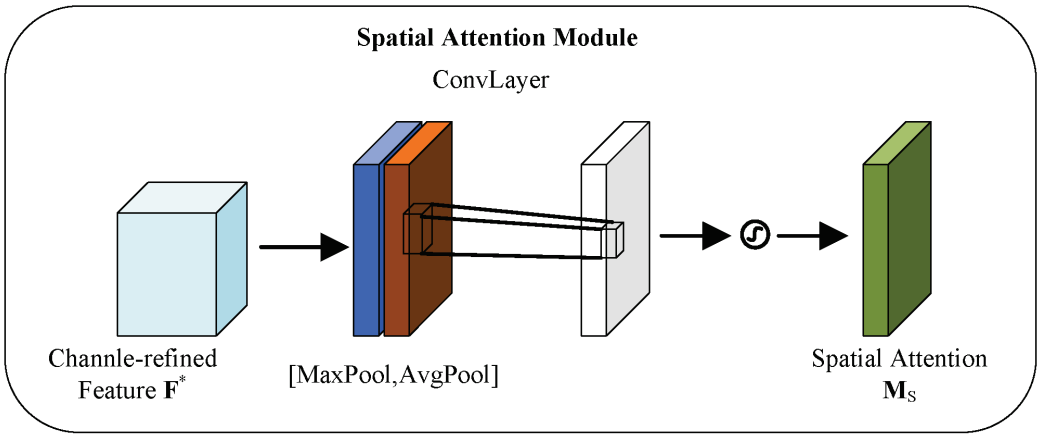


Figure 5. Spatial attention module diagram.

The specific structure of the spatial attention module is shown in Figure 5. The feature maps processed by the channel attention module first pass through two convolution layers to obtain the weight coefficients of each pixel point. Here, the weight coefficients are calculated by considering the response of each pixel point on different spatial scales, and the similarity between pixel points. Finally, the weight coefficients are applied to the feature map to emphasize the important pixel points and suppress the unimportant ones. This improves the generalization ability of the model. The spatial attention module focuses on which location information is meaningful, and it also complements the channel attention module, as shown below:

$$M_C(F) = \sigma(\text{Conv}[\text{AvgPool}(F); \text{MaxPool}]). \tag{12}$$

where $\sigma(\cdot)$ denotes the Sigmoid function, MaxPool$(\cdot)$ denotes the maximum pooling, AvgPool$(\cdot)$ denotes the average pooling, MLP$(\cdot)$ denotes the multilayer perceptron, and Conv$(\cdot)$ denotes the 3D convolutional layer.

Finally, the model uses the Softmax layer to calculate the final traffic class using the output of the residual module and the CBAM module. The outputs of the residual and CBAM modules and the raw inputs are fed into the fully connected layer after manipulation. The fully connected layer then multiplies the weight matrix with the input vector and adds the bias as follows:

$$z_j = w_j \cdot X + b_j = w_{j1}X_1 + w_{j2}X_2 + \cdots + w_{jn}X_n + b_j. \tag{13}$$

where X is the input of the fully connected layer, w_j is the weight of the j-th class of features, and b_j is the bias term. The feedforward layer represents the captured spatiotemporal features as a linear representation suitable for predicting the final category labels using Softmax operations. In this process, each category is assigned a probability score, and

the category with the highest probability is considered the final model prediction, as shown below:

$$p = \text{SoftMax}(z_j) = \frac{e^{z_j}}{\sum_K e^{z_j}}. \tag{14}$$

$$\hat{y} = \arg\max(p). \tag{15}$$

where z_j denotes the output of the feedforward layer and p denotes the probability score. The training model is calculated to minimize the cross-entropy loss according to Equation (16):

$$\text{Loss} = -\sum(y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log(1 - \hat{y}_i)). \tag{16}$$

where y_i denotes the actual label and $\hat{y}_i$ denotes the model prediction label.

To maximize feature reuse capability, this study uses jump connections to add the output of the previous parameter layer to its subsequent parameter layers to maintain local originality throughout the learning phase. In addition, each pair of jump-connected residual and attention modules will be connected to all subsequent pairs of residual and attention modules to maintain global originality throughout the learning phase. The proposed network structure has several significant advantages to improve its generalization performance.

6. Model Evaluation and Discussions

6.1. Dataset Description

Generalization ability is an important metric to assess the performance of a model. A good model should have strong generalization ability, meaning it can perform well on new datasets. Therefore, in this study, the model's generalization performance is tested using different datasets.

To evaluate the performance of the proposed model, we used ToN_IoT [13], a real dataset created from a large-scale IoT system developed by the Cyber IoT Laboratory at ADFA, New South Wales, Australia. This dataset contains normal category data and nine categories of attack data, including password, scan, ransomware, backdoor, denial of service, distributed denial of service, MITM, injection, and XSS attacks. The total number of data instances in this dataset is shown in Table 1. We trained our model using the ToN_IoT dataset to ensure it learns from a comprehensive set of attacks and normal behavior data.

Table 1. ToN_IoT dataset record description.

Record Type	Number of Records
Backdoor	35,000
DDoS	25,000
DoS	20,000
Injection	35,000
MITM	1043
Password	35,000
Ransomware	16,030
Scanning	3973
XSS	6116
Normal	245,000

To test the generalization ability of our model, we then used the UNSW-NB15 [31] dataset. The raw network packets for the UNSW-NB15 dataset are created by the IXIA PerfectStorm tool at the Australian Cyber Security Centre's Cyber Scope Lab, which can be used to generate a mix of modern normal activity and synthetic contemporary attack behavior. The dataset contains a total of 49 features and 9 common attacks. The normal information accounts for 88% of the dataset size and the attack information accounts for 12%. We used this dataset to test the generalization ability of the model. The basic information of this dataset is shown in Table 2.

Table 2. UNSW-NB15 dataset record description.

Record Type	Number of Records
Analysis	2000
Backdoor	1746
DoS	12,264
Exploits	33,393
Fuzzers	18,184
Generic	40,000
Reconnaissance	10,491
Shellcode	1133
Worms	130
Normal	56,000

We use several metrics to evaluate the performance of the proposed model, i.e., *accuracy*, *precision*, *recall*, and *F1-score* because they are widely used to evaluate deep learning algorithms. The formula is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{17}$$

$$Precision = \frac{TP}{TP + FP} \tag{18}$$

$$Recall = \frac{TP}{TP + FN} \tag{19}$$

$$F1\text{-}score = \frac{2TP}{2TP + FP + FN} \tag{20}$$

where TP is true positive, representing the number of positive samples correctly identified as positive, FP is false positive, representing the number of negative samples incorrectly identified as positive, TN is true negative, representing the number of negative samples correctly detected as negative, and FN is false negative, representing the number of positive samples incorrectly identified as negative.

6.2. Hyperparameter Settings

Hyperparameters are parameters that are artificially adjusted before or during training. To ensure the best performance of the proposed deep learning model, optimal hyperparameters are determined through extensive experimentation. The hyperparameters involved in the experimental process include learning rate, epoch, and batch size. The details are shown in Table 3.

- Learning rate (LR): A critical hyperparameter in deep learning that regulates the network model's learning progress. In this paper, we set three learning rate values to obtain the best performance.
- Batch size: This parameter indicates the number of samples selected for one training. The batch size affects the memory usage, as well as the optimization and speed of the model. The batch size cannot be set too large or too small, so we set the range of batch size to 16, 32, 64, 128, 256, 512, 1024.
- Epoch: The epoch represents the process of training all the training samples once. Too many epochs can lead to overfitting, while too few epochs may result in suboptimal training parameters. In this study, the number of epochs is set to 100.

We divided the ToN_IoT dataset into a training set and a test set by the ratio of 80% and 20%, respectively. Initially, we set the learning rate to 0.01, epochs fixed to 100 times, and batch size range set to 16, 32, 64, 128, 256, 512, and 1024. We conducted experiments at different batch sizes and recorded the results for the above learning rates in Table 3. A detailed performance comparison of these parameters is shown in Figure 6a. The best-

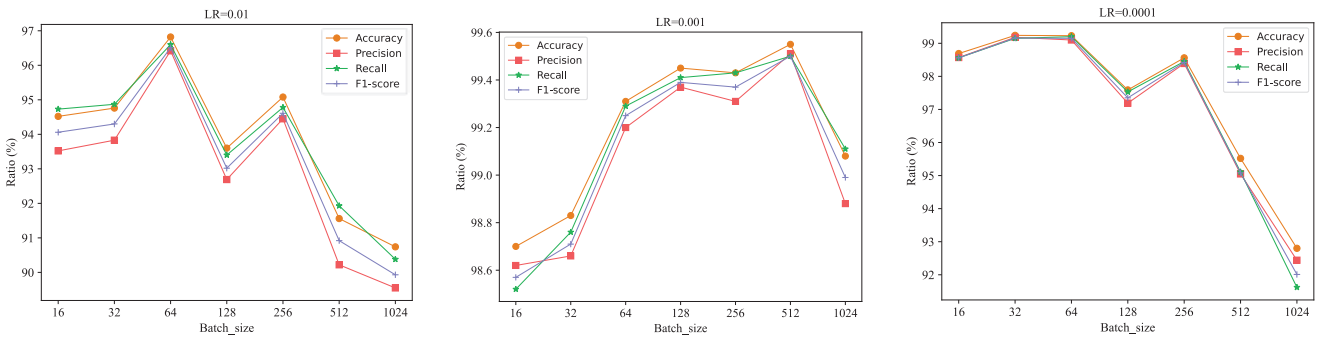
tested accuracy of the model is 96.82% for a learning rate of 0.01 and a batch size of 64. The lowest value of accuracy for this learning rate is 90.74% for a batch size of 1024. For other batch sizes, the accuracy and other performance scores are above 90%, demonstrating the model's robustness to changes in batch size.

Table 3. Performance variation at different parameters.

Performance Parameters		Batch Size						
		16	**32**	**64**	**128**	**256**	**512**	**1024**
LR = 0.01	Accuracy	0.9452	0.9476	**0.9682**	0.9360	0.9508	0.9156	0.9074
	Precision	0.9352	0.9383	**0.9642**	0.9269	0.9445	0.9022	0.8955
	Recall	0.9473	0.9487	**0.9660**	0.9340	0.9478	0.9193	0.9038
	F1-score	0.9406	0.9430	**0.9651**	0.9302	0.9461	0.9092	0.8993
LR = 0.001	Accuracy	0.9870	0.9883	0.9931	0.9945	0.9943	**0.9955**	0.9908
	Precision	0.9862	0.9866	0.9920	0.9937	0.9931	**0.9951**	0.9888
	Recall	0.9852	0.9876	0.9929	0.9941	0.9943	**0.9950**	0.9911
	F1-score	0.9857	0.9871	0.9925	0.9939	0.9937	**0.9950**	0.9899
LR = 0.0001	Accuracy	0.9869	**0.9924**	0.9923	0.9759	0.9856	0.9552	0.9280
	Precision	0.9857	**0.9918**	0.9910	0.9719	0.9839	0.9505	0.9244
	Recall	0.9856	**0.9915**	0.9920	0.9753	0.9845	0.9511	0.9162
	F1-score	0.9856	**0.9916**	0.9915	0.9735	0.9842	0.9508	0.9201

In the second stage, we fixed the learning rate at 0.001 and the epoch remained fixed at 100 times, while the batch sizes ranged from 16, 32, 64, 128, 256, 512, and 1024. By keeping the learning rate and epoch fixed, we conducted experiments at different batch sizes and recorded the results in Table 3. A detailed performance comparison of these parameters is shown in Figure 6b. The best test accuracy is 99.55% when the batch size is 512. At batch sizes of 64, 128, 256, and 1024, the accuracy of the model was greater than 99%. At lower batch sizes of 16 and 32, the accuracy and other performance scores decreased but were still close to 99%. This indicates that a learning rate of 0.001 with a batch size of 512 provides the optimal balance between training stability and performance.

Finally, we set the learning rate to 0.0001, and the other parameters were kept the same as those in the first two stages. The detailed performance comparison of these parameters is shown in Figure 6c. The best test accuracy is 99.24% when the batch size is 32. For batch sizes of 16, 64, and 256, the accuracy of the model is above 98%. For other batch sizes, the accuracy and other performance scores decreased but the results were above 90%. The comparative experimental results of the above performance parameters show that the proposed model achieves the best results when the learning rate is 0.001, epochs are 100, and the batch size is 512. This combination of parameters consistently provides high accuracy and stable performance across different configurations, demonstrating the robustness and generalizability of the proposed model.



(**a**) LR = 0.01 performance. (**b**) LR = 0.001 performance. (**c**) LR = 0.0001 performance.

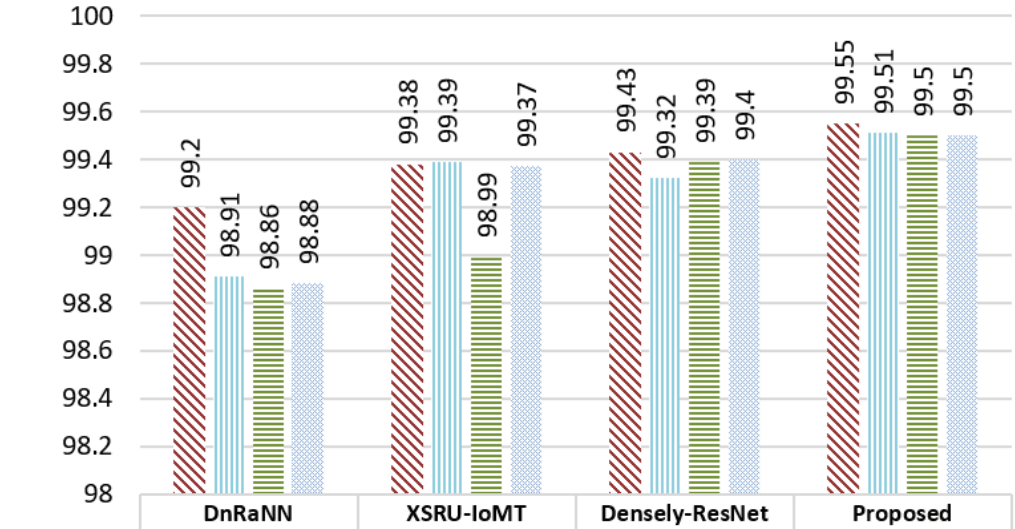
Figure 6. Performance at different learning rates.

Finally, we set the learning rate to 0.0001, and the other parameters were kept the same as those in the first two stages. The detailed performance comparison of these parameters is shown in Figure 6c. The best test accuracy is 99.24% when the batch size is 32. For batch sizes of 16, 64, and 256, the accuracy of the model is above 98%. For other batch sizes, the accuracy and other performance scores decreased, but the results were above 90%.

The comparative experimental results of the above performance parameters show that the proposed model achieves the best results when the learning rate is 0.001, epochs are 100, and batch size is 512.

6.3. Comparison with State-of-the-Art Methods

To further analyze the effectiveness of the proposed model, we compared the proposed model with some of the best current methods and evaluated the performance differences between the proposed model and other methods. The result is shown in Figure 7. As can be seen from the figure, the proposed model has the best performance on the ToN_IoT dataset, outperforming existing models in terms of accuracy, precision, recall, and F1-score. Specifically, the proposed model achieved an increase in accuracy from 99.20% with DnRaNN to 99.55%, an increase of 0.35%. Compared to XSRU-IoMT, the accuracy increased from 99.38% to 99.55%, which is an increase of 0.17%. Additionally, the accuracy improved from 99.43% with Densely-ResNet to 99.55%, an increase of 0.12%. These results verify the validity of our proposed model.



	DnRaNN	XSRU-IoMT	Densely-ResNet	Proposed
Accuracy	99.2	99.38	99.43	99.55
Precision	98.91	99.39	99.32	99.51
Recall	98.86	98.99	99.39	99.5
F1-score	98.88	99.37	99.4	99.5

Figure 7. Performance comparison with other models.

In addition, we compared the generalization ability of the model with several deep learning methods on the UNSW-NB15 dataset, as shown in Table 4. Our proposed LSTM-ResNet model achieved accuracy, precision, recall, and F1-score values of 89.23%, 88.83%, 87.77%, and 88.25%, respectively. This represents a significant improvement in performance metrics compared to other models. For instance, the accuracy of our model is 15.3% higher than that of the Densely-ResNet model, which is the second-best performing model in our comparison.

The severe class imbalance problem in the UNSW-NB15 benchmark often leads to poor model generalization performance. However, our proposed model demonstrates robust performance despite this challenge. As shown in Table 4, the LSTM-ResNet model

not only achieves the highest accuracy but also excels in precision, recall, and F1-score compared to LSTM, LuNet, Densely-ResNet, and CNN models. This indicates that our model has superior generalization capability and can effectively handle imbalanced data. These results highlight the robustness and reliability of our method in diverse scenarios, confirming its effectiveness in real-world applications.

Moreover, the attention mechanism incorporated in our model helps it focus on important features, further enhancing its performance and robustness. This makes the LSTM-ResNet model particularly effective in identifying and classifying network intrusions, as evidenced by its leading performance across multiple evaluation metrics.

Table 4. Comparison of model generalization performance on the UNSW-NB15 dataset.

Model	Accuracy	Precision	Recall	F1-Score
LSTM	0.7015	0.7754	0.7724	0.8628
LuNet	0.7267	0.7850	0.8290	0.8750
Densely-ResNet	0.7393	0.8094	0.8668	0.8811
CNN	0.8163	0.8094	0.8578	0.8121
LSTM-ResNet	**0.8923**	**0.8883**	**0.8777**	**0.8825**

7. Conclusions

In this paper, we propose a deep learning model based on the temporal convolution residual module and attention mechanism for IoT anomaly detection and analyze the model in depth on the ToN_IoT dataset, which is the latest publicly available IoT dataset. In addition, the proposed model is evaluated on the UNSW-NB15 dataset, and its generalization performance is compared with several deep learning methods. As a result, ResNet achieves state-of-the-art detection accuracy on the UNSW-NB15 benchmarks while maintaining a low false positive rate. The evaluation results confirm the effectiveness and stability of the proposed model, and it is recommended for use in other intrusion detection tasks in the future.

Author Contributions: Conceptualization, B.C. and Y.C.; methodology, B.C. and Y.C.; investigation, B.C. and Y.C.; writing—original draft preparation, B.C., Y.C., K.L. and Z.Y.; writing—review and editing, B.C., Y.C., K.L. and Z.Y.; visualization, Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This paper is supported by the National Natural Science Foundation of China (61962042), the Natural Science Foundation of Inner Mongolia (2022MS06020), the Central Government Guides Local Science and Technology Development Fund (2022ZY0064), the University Youth Science and Technology Talent Development Project (Innovation Group Development Plan) of Inner Mongolia A. R. of China (grant no. NMGIRT2318), and the Fund for Supporting the Reform and Development of Local Universities (Disciplinary Construction).

Data Availability Statement: The datasets used in this study are publicly available at the following links: https://research.unsw.edu.au/projects/unsw-nb15-dataset and https://research.unsw.edu.au/projects/toniot-datasets, accessed on 1 May 2024.

Acknowledgments: The authors would like to extend their sincere appreciation to the reviewers for their invaluable feedback and insightful suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bertino, E.; Islam, N. Botnets and internet of things security. *Computer* **2017**, *50*, 76–79. [CrossRef]
2. Kolias, C.; Kambourakis, G.; Stavrou, A.; Voas, J. DDoS in the IoT: Mirai and other Botnets. *Computer* **2017**, *50*, 80–84. [CrossRef]
3. Abomhara, M.; Koien, G.M. Cyber security and the internet of things: Vulnerabilities, threats, intruders and attacks. *J. Cyber Secur. Mobil.* **2015**, *4*, 65–88. [CrossRef]

4. Thakkar, A.; Lohiya, R. A review on machine learning and deep learning perspectives of IDS for IoT: Recent updates, security issues, and challenges. *Arch. Comput. Methods Eng.* **2021**, *28*, 3211–3243. [CrossRef]
5. Al-Garadi, M.A.; Mohamed, A.; Al-Ali, A.K.; Du, X.; Ali, I.; Guizani, M. A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE Commun. Surv. Tutorials* **2020**, *22*, 1646–1685. [CrossRef]
6. Babu, M.R.; Veena, K.N. A survey on attack detection methods for IoT using machine learning and deep learning. In Proceedings of the 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 625–630.
7. Denning, D.E. An intrusion-detection model. *IEEE Trans. Softw. Eng.* **1987**, *SE-13*, 222–232. [CrossRef]
8. Chaabouni, N.; Mosbah, M.; Zemmari, A.; Sauvignac, C.; Faruki, P. Network intrusion detection for IoT security based on learning techniques. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2671–2701. [CrossRef]
9. Alsoufi, M.A.; Razak, S.; Siraj, M.M.; Nafea, I.; Ghaleb, F.A.; Saeed, F.; Nasser, M. Anomaly-based intrusion detection systems in IoT using deep learning: A systematic literature review. *Appl. Sci.* **2021**, *11*, 8383. [CrossRef]
10. Selvapandian, D.; Santhosh, R. Deep learning approach for intrusion detection in IoT-multi cloud environment. *Autom. Softw. Eng.* **2021**, *28*, 19. [CrossRef]
11. Khoa, T.V.; Saputra, Y.M.; Hoang, D.T.; Trung, N.L.; Nguyen, D.; Ha, N.V.; Dutkiewicz, E. Collaborative learning model for cyberattack detection systems in IoT industry 4.0. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Republic of Korea, 25–28 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–6.
12. Haider, A.; Adnan Khan, M.; Rehman, A.; Rahman, M.; Kim, S.H. A real-time sequential deep extreme learning machine cybersecurity intrusion detection system. *Comput. Mater. Contin.* **2021**, *66*, 1785–1798. [CrossRef]
13. Booij, T.M.; Chiscop, I.; Meeuwissen, E.; Moustafa, N.; Den Hartog, F.T. ToN_IoT: The role of heterogeneity and the need for standardization of features and attack types in IoT network intrusion data sets. *IEEE Internet Things J.* **2021**, *9*, 485–496. [CrossRef]
14. Tsimenidis, S.; Lagkas, T.; Rantos, K. Deep learning in IoT intrusion detection. *J. Netw. Syst. Manag.* **2022**, *30*, 8. [CrossRef]
15. Wu, P.; Moustafa, N.; Yang, S.; Guo, H. Densely connected residual network for attack recognition. In Proceedings of the 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Guangzhou, China, 29 December 2020–1 January 2021; IEEE: Piscataway, NJ, USA, 2020; pp. 233–242.
16. Hasan, M.; Islam, M.M.; Zarif, M.I.I.; Hashem, M.M.A. Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. *Internet Things* **2019**, *7*, 100059. [CrossRef]
17. Ravi, N.; Shalinie, S.M. Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture. *IEEE Internet Things J.* **2020**, *7*, 3559–3570. [CrossRef]
18. Zhang, B.; Liu, Z.; Jia, Y.; Ren, J.; Zhao, X. Network intrusion detection method based on PCA and Bayes algorithm. *Secur. Commun. Netw.* **2018**, *2018*, 1914980. [CrossRef]
19. Abdel-Basset, M.; Chang, V.; Hawash, H.; Chakrabortty, R.K.; Ryan, M. Deep-IFS: Intrusion detection approach for industrial internet of things traffic in fog environment. *IEEE Trans. Ind. Inform.* **2020**, *17*, 7704–7715. [CrossRef]
20. Liu, Y.; Garg, S.; Nie, J.; Zhang, Y.; Xiong, Z.; Kang, J.; Hossain, M.S. Deep anomaly detection for time-series data in industrial IoT: A communication-efficient on-device federated learning approach. *IEEE Internet Things J.* **2020**, *8*, 6348–6358. [CrossRef]
21. Li, L.; Yan, J.; Wang, H.; Jin, Y. Anomaly detection of time series with smoothness-inducing sequential variational auto-encoder. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 1177–1191. [CrossRef] [PubMed]
22. Gao, J.; Gan, L.; Buschendorf, F.; Zhang, L.; Liu, H.; Li, P.; Dong, X.; Lu, T. Omni SCADA intrusion detection using deep learning algorithms. *IEEE Internet Things J.* **2020**, *8*, 951–961. [CrossRef]
23. Parra, G.D.L.T.; Rad, P.; Choo, K.K.R.; Beebe, N. Detecting internet of things attacks using distributed deep learning. *J. Netw. Comput. Appl.* **2020**, *163*, 102662.
24. Khan, I.A.; Moustafa, N.; Razzak, I.; Tanveer, M.; Pi, D.; Pan, Y.; Ali, B.S. XSRU-IoMT: Explainable simple recurrent units for threat detection in internet of medical things networks. *Future Gener. Comput. Syst.* **2020**, *127*, 181–193. [CrossRef]
25. Wu, P.; Guo, H. LuNET: A deep neural network for network intrusion detection. In Proceedings of the 2019 IEEE Symposium Series on Computational Intelligence (SSCI), Xiamen, China, 6–9 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 617–624.
26. Latif, S.; e Huma, Z.; Jamal, S.S.; Ahmed, F.; Ahmad, J.; Zahid, A.; Dashtipour, K.; Aftab, M.U.; Ahmad, M.; Abbasi, Q.H. Intrusion detection framework for the internet of things using a dense random neural network. *IEEE Trans. Ind. Inform.* **2021**, *18*, 6435–6444. [CrossRef]
27. Ferrag, M.A.; Maglaras, L.; Moschoyiannis, S.; Janicke, H. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *J. Inf. Secur. Appl.* **2020**, *50*, 102419. [CrossRef]
28. Guo, M.H.; Xu, T.; Liu, J.J.; Liu, Z.N.; Jiang, P.T.; Mu, T.J.; Zhang, S.H.; Martin, R.; Cheng, M.M.; Hu, S.M. Attention mechanisms in computer vision: A survey. *Comput. Vis. Media* **2022**, *8*, 331–368. [CrossRef]
29. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 770–778.

30. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional block attention module. In Proceedings of the 2018 European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 3–19.
31. Moustafa, N.; Slay, J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia, 10–12 November 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–6.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Article

A Novel Hybrid Unsupervised Learning Approach for Enhanced Cybersecurity in the IoT

Prabu Kaliyaperumal [1,*], Sudhakar Periyasamy [1], Manikandan Thirumalaisamy [2], Balamurugan Balusamy [3] and Francesco Benedetto [4,*]

[1] School of Computer Science and Engineering, Galgotias University, Dankaur 203201, India; p.sudhakar@galgotiasuniversity.edu.in
[2] Department of CSBS, Rajalakshmi Engineering College, Tamil Nadu 602105, India; tmcse1404@gmail.com
[3] Associate Dean-Students, Shiv Nadar University, Delhi-NCR Campus, Noida 201305, India; kadavulai@gmail.com
[4] Signal Processing for TLC and Economics, University of Roma Tre, 00154 Rome, Italy
* Correspondence: k.prabu@galgotiasuniversity.edu.in (P.K.); francesco.benedetto@uniroma3.it (F.B.)

Abstract: The proliferation of IoT services has spurred a surge in network attacks, heightening cybersecurity concerns. Essential to network defense, intrusion detection and prevention systems (IDPSs) identify malicious activities, including denial of service (DoS), distributed denial of service (DDoS), botnet, brute force, infiltration, and Heartbleed. This study focuses on leveraging unsupervised learning for training detection models to counter these threats effectively. The proposed method utilizes basic autoencoders (bAEs) for dimensionality reduction and encompasses a three-stage detection model: one-class support vector machine (OCSVM) and deep autoencoder (dAE) attack detection, complemented by density-based spatial clustering of applications with noise (DBSCAN) for attack clustering. Accurately delineated clusters aid in mapping attack tactics. The MITRE ATT&CK framework establishes a "Cyber Threat Repository", cataloging attacks and tactics, enabling immediate response based on priority. Leveraging preprocessed and unlabeled normal network traffic data, this approach enables the identification of novel attacks while mitigating the impact of imbalanced training data on model performance. The autoencoder method utilizes reconstruction error, OCSVM employs a kernel function to establish a hyperplane for anomaly detection, while DBSCAN employs a density-based approach to identify clusters, manage noise, accommodate diverse shapes, automatically determining cluster count, ensuring scalability, and minimizing false positives and false negatives. Evaluated on standard datasets such as CIC-IDS2017 and CSECIC-IDS2018, the proposed model outperforms existing state of art methods. Our approach achieves accuracies exceeding 98% for the two datasets, thus confirming its efficacy and effectiveness for application in efficient intrusion detection systems.

Keywords: autoencoder; DBSCAN; support vector machine; unsupervised learning; cloud security

1. Introduction

The continuous growth of the Internet has led to numerous cybersecurity challenges, necessitating robust solutions to protect digital infrastructures. Intrusion detection systems (IDSs) play a crucial role by meticulously analyzing data streams such as network packets for signs of threats [1]. Positioned as the second line of defense after firewalls, as illustrated in Figure 1, IDSs utilize advanced algorithms to identify anomalies, facilitating prompt response from administrators. Beyond threat detection, Figure 2 illustrates that IDSs offer real-time monitoring, detailed analysis, instant alerts, and administrative controls, enabling organizations to stay vigilant against evolving cyber threats. In essence, an IDS is an essential component of modern cybersecurity, fortifying networks against relentless attacks in an era of escalating digital risks [2]. The proposed system architecture showcases col-

laborative efforts enhancing cybersecurity, with machine learning (ML) and deep learning (DL) pivotal in anomaly detection.

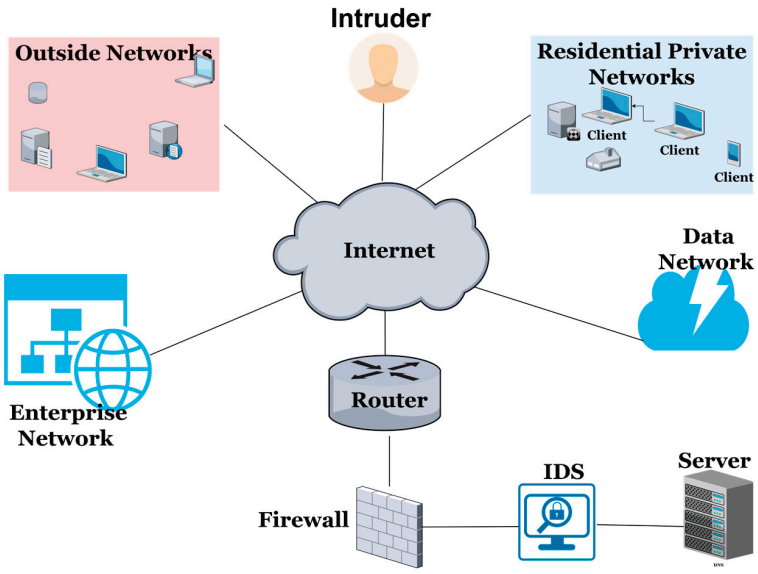


Figure 1. Operating principles of Intrusion Detection System.

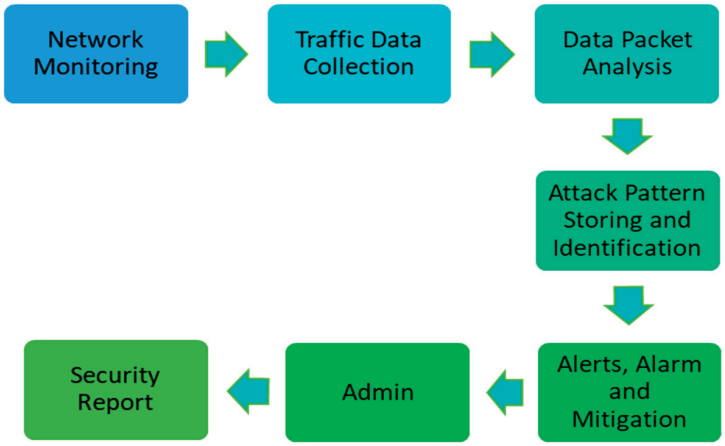


Figure 2. Elements of intrusion detection and prevention system.

1.1. Motivation

In 2023, Douglas AutoTech Corp, Hopkinsville, KY, USA, a prominent automotive manufacturer, upgraded its entire production line to utilize IoT technology in a smart factory setup. This transformation was intended to boost efficiency and lower expenses by linking machinery, sensors, and production systems into a centralized IoT network. While the transition yielded substantial operational improvements, it also introduced several cybersecurity vulnerabilities. AutoTech Corp soon found that its IoT network had become a prime target for malicious actors using increasingly sophisticated tactics, prompting the need for strong cybersecurity measures to protect its operations.

The Bitdefender 2023 IoT Security Landscape Report underscores a significant event in 2023: a large-scale DDoS attack targeting a smart home network. The attack took advantage of weaknesses in different IoT devices, including security cameras, smart thermostats, and connected appliances. This attack interrupted the regular functioning of these devices, causing considerable inconvenience and posing a potential security threat to users. The attackers employed a botnet made up of compromised IoT devices to initiate the surge of malicious traffic. The report provides details on multiple attack vectors such as brute

force, spoofing, and web-based attacks. These tactics are becoming increasingly common as the number of connected devices continues to rise. These incidents highlight the urgent requirement for strong security measures and ongoing monitoring in IoT environments to protect against emerging cyber threats.

1.2. Problem Statement

Supervised models are proficient at identifying known attacks; however, they depend significantly on the attack patterns learned during their training phase. Consequently, they find it challenging to detect unknown or emerging attacks when such threats manifest in network traffic. Special attention is devoted to novel attacks commonly known as zero-day attacks and false negatives, which can lead to undetected security breaches, heightened vulnerability, and damage to reputation [3,4]. To overcome this limitation, we have implemented an unsupervised learning approach that can detect unknown, emerging, or zero-day attack vectors. This approach, which is trained on imbalanced and unlabeled data, is capable of identifying abnormal behavior in network traffic without the need for pre-labeled attack patterns. Unsupervised learning models provide significant benefits for IDSs, especially in situations where labeled data are limited or unavailable. Unlike supervised learning, which relies on labeled datasets for training, unsupervised learning can function effectively using unlabeled data. This is particularly advantageous in the context of IDS, where acquiring a complete and precisely labeled dataset encompassing all possible attack types can be difficult and expensive.

Leveraging these techniques strengthens cybersecurity, enabling real-time risk detection and mitigation. The quality of training data profoundly affects unsupervised anomaly detection's efficacy, emphasizing the need for diverse datasets to address traffic feature variations [5–8]. Neglecting complex data patterns can impair detection. Addressing these challenges enhances system robustness, aiding proactive cybersecurity measures against emerging threats in the digital realm.

1.3. Expected Outcomes

The proposed method initiates by employing dimensionality reduction through a basic autoencoder (bAE). This technique aims to extract encoded features from individual data categories, enhancing the efficiency of subsequent processing and analysis [9]. In its role as a stage 1 detector, the one-class support vector machine (OCSVM) utilizes the hyperparameter ν to regulate the trade-off, acting as an upper limit on the proportion of anomalies included in model training from normal network traffic.

The kernel function is designed to establish a hyperplane that effectively segregates data from the origin [10,11]. Anomalies are identified when new samples deviate to the opposite side of the hyperplane, thereby yielding a negative value in the decision function for these outlier points [12].

The OCSVM accurately classifies normal traffic, supplying essential data for the dAE neural network, forming the foundation of a stage 2 detector for effective anomaly detection. The dAE captures essential features for model training from normal network traffic. By computing the disparity between input and output, a detection threshold is set, ensuring precise discrimination between normal and abnormal traffic [13].

After passing through the OCSVM and dAE, abnormal traffic undergoes processing with DBSCAN, which serves as a stage 3 detector, forming clusters based on feature density. DBSCAN's key parameters, such as epsilon (ε) and MinPts, influence cluster shape and density, providing the ability to identify dense clusters, handle noise, adapt to various shapes, automatically determine cluster count, scale effectively, and reduce false negatives and false positives [14]. These accurately delineated clusters can help in mapping attack tactics for intrusion prevention. Using the MITRE ATT&CK framework [15], we establish the "Cyber Threat Repository" to catalog attacks, tactics, and mitigation strategies. Detected attacks trigger immediate responses based on priority levels.

The proposed methodology focuses on employing OCSVM + dAE + DBSCAN for outlier detection. It entails analyzing the disparity between the input and output of the detection model and assessing density within the clustering model. Our novel approach involves reducing dimensions, training models, detecting anomalies, clustering them, and ultimately preventing any identified anomalies.

During dimensionality reduction, the basic autoencoder (bAE) extracts features and captures normal traffic data without supervision, resulting in unique patterns within the latent space features. In one-class SVMs, the hyperparameter "nu" (ν) governs the balance between capturing normal instances and minimizing margin violations or support vectors [10]. In anomaly detection, when test data are input, samples that fall on the wrong side of the hyperplane are identified as anomaly points. Anomalous points are determined by a decision function that yields a negative value [16]. The dAE neural network proceeds to reconstruct the input data via encoding–decoding, generating an average reconstruction error employed as the detection threshold [17]. The output of normal traffic from the OCSVM serves as input, and the mean squared error (MSE) is calculated to determine the reconstruction error. If this error surpasses the established threshold, the traffic is flagged as abnormal; otherwise, it is classified as normal [18]. Following this, abnormal traffic from both OCSVM and dAE is subjected to DBSCAN processing, which identifies clusters based on packet feature density. The critical parameters of DBSCAN, epsilon (ε) and MinPts, affect both the shape and density of the clusters [14]. Upon detection of an attack by the intrusion detection module, the intrusion prevention module is triggered to mitigate the intrusion and take necessary actions. Integrated with the MITRE ATT&CK threat intelligence framework [19], the prevention module implements resolution based on priority if the attack is already in the cyber threat repository database. However, for new attacks not in the database, cybersecurity personnel address the issue accordingly.

1.4. Key Contributions

1. We introduce a novel hybrid algorithm rooted in deep learning principles. It initiates with raw traffic data, utilizing autoencoder for dimensionality reduction.
2. Our hybrid algorithm employs a one-class SVM and a deep autoencoder neural network to distinguish between non-attack and attack data. Identified attacks undergo processing using DBSCAN to form precise attack clusters.
3. Upon detection of attacks, the MITRE ATT&CK-based prevention system is activated to respond accordingly.
4. Validation entails assessing enhanced metrics such as accuracy, precision, specificity, recall, F-measure, false negative rate (FNR), prevention rate, priority-based blocking rate, and success rate on two datasets, namely, CSECIC-IDS2018 and CIC-IDS2017.

This research article unfolds as follows: Section 1 covers intrusion detection basics and prerequisites. Section 2 explores the current literature. Section 3 describes the datasets utilized, Section 4 details our novel method, while Section 5 scrutinizes the results as well as the comparative analysis with state-of-art methods. Finally, Section 6 concludes our paper, offering insights and suggesting future research directions.

2. Related Works

Intrusion detection systems (IDSs) are classified into host-based (HIDS) and network-based (NIDS) types. An HIDS analyzes host data like operating system logs, while an NIDS monitors network traffic for detecting malicious activities [20]. This study focuses on NIDSs, due to their crucial role in network security. Supervised IDS methods, necessitating labeled datasets, face challenges with deep learning's data hunger, especially in imbalanced data scenarios. A solution is the semi-supervised anomaly-based NIDS, trained solely on normal data [1,21]. This method quantifies abnormality levels, thus identifying potential anomalies by elevated anomaly scores, enhancing adaptability and effectiveness in threat mitigation.

Supervised learning in abnormal traffic detection deploys machine learning algorithms to classify data, identifying and flagging unusual patterns in the dataset [22,23]. Several

researchers [24] have developed NIDSs based on machine learning and deep learning, owing to their impressive performance.

Fan et al. in [25] presented the RF-SVM-IL framework for detecting DDoS attacks, utilizing random forest (RF) and support vector machine (SVM) for classifying dual traffic data. They incorporated an incremental learning (IL) algorithm to sift through extra input samples, thereby lessening computational burden and improving the model's capacity to precisely categorize traffic, especially under high attack volumes. However, the article notes that while SVMs can manage high-dimensional data, their training space and computational complexity increase significantly with larger datasets. This results in limited processing capacity for handling large-scale data.

Yaras S et al. [26] introduces a hybrid deep learning algorithm that integrates convolutional neural network (CNN) and long short-term memory (LSTM) models for detecting DDoS attacks. The methodology includes preprocessing and feature selection on the CICIoT2023 and TON_IOT datasets, followed by testing the algorithm in both binary and multiclass scenarios. The dataset comprises diverse IoT attacks collected from real-world environments. The algorithm achieved impressive accuracy rates: 99.995% for detecting attacks and 99.96% for identifying attack types with CICIoT2023, and 98.75% with TON_IOT. This approach offers advantages by harnessing the capabilities of both CNNs and LSTM, resulting in superior detection accuracy and a minimal false positive rate. However, employing extensive datasets amplifies both training and testing durations. Future avenues include refining algorithm parameters through metaheuristic techniques and streamlining training processes to build efficient, cost-effective intrusion detection systems. The study makes a substantial contribution to the literature by establishing a benchmark for future research in the field.

Harahsheh K et al. [27] presents a hybrid feature selection approach tailored for IoT environments, utilizing the InSDN dataset. The approach includes multiple stages: preprocessing the data, reducing dimensions, selecting features, evaluating models, and incorporating a caching mechanism to improve efficiency. Random forest is utilized for both ranking features and detecting anomalies. The study bases its analysis on the InSDN dataset, which comprises 343,889 records of normal and various attack traffic. The method achieves exceptionally high accuracy (99.99%) at a minimal computational cost (0.8599 s) and effectively reduces the feature set from 84 to 11, making it well suited for IoT environments with limited resources. However, the method's dependence on stable feature counts for caching could be problematic if column content changes while the feature count remains constant. Future research will aim to investigate the data within columns to overcome this limitation and further enhance feature selection methods for improved accuracy and efficiency.

Javed A et al. [28] introduces a holistic approach to enhance intrusion detection systems (IDSs) in IoT environments. The authors employed the TON_IoT dataset, which consists of 461,043 instances and encompasses various attack categories, simulating behaviors at the edge, fog, and cloud levels. The methods included feature scaling, label encoding, chi-square for feature selection, and principal component analysis (PCA) for feature extraction. The random forest algorithm was selected for its robustness in handling continuous and categorical data, its capability to manage missing values effectively, and its ability to parallelize computations. The proposed model attained a 99.99% accuracy with a computational cost of 0.8599 s, utilizing only 11 features. Benefits include achieving high accuracy and efficiency in environments with limited resources. However, the approach's applicability to diverse environmental conditions is restricted. Future research aims to enhance the methodology by delving into column-specific data analysis, broadening the range of attack types, and extracting additional network parameters from IoT devices.

Liao et al. in [29] devised an ensemble framework incorporating various autoencoders (AEs) and generative adversarial networks (GANs). The framework integrates traditional AE (tAE), variational AE (vAE), convolutional AE (cAE), convolutional variational AE (cvAE), and a GAN. A weighted average ensemble aggregates anomaly scores derived from multiple models' reconstruction errors post-training. By comparing scores to a predefined

threshold, samples are classified. Experimental results showcase superior performance of the ensemble model over single models, prompting the collaborative use of multiple detectors for anomaly identification. However, handling the complexity and computational resources needed for ensemble methods can pose challenges.

Almaraz-Rivera J.G. et al. [30] introduces a network-based intrusion detection system (IDS) designed to safeguard IoT networks against distributed denial-of-service (DDoS) attacks. The methodology includes generating synthetic images using flow-level traffic data from the LATAM-DDoS-IoT and Bot-IoT datasets and exploring both supervised and self-supervised learning approaches. The findings reveal that in specific tests, self-supervised learning outperforms supervised learning, demonstrating a 4.83% improvement in F1-measure and a 14.61% increase in accuracy for multiclass protocol classification. The article emphasizes the benefits of self-supervised learning, including its ability to eliminate the requirement for extensive labeled data and its capacity for robust generalization. However, it also acknowledges the challenges in generalizing, particularly in dynamic threat landscapes with emerging attack types. Future research directions aim to refine training frameworks for contrastive learning experiments and to delve deeper into visual representations within cybersecurity domains.

Ansam Khraisat et al. [31] introduce a novel hybrid intrusion detection system that integrates the C5.0 decision tree classifier and the one-class support vector machine (OC-SVM) to improve detection accuracy and minimize false alarms. This hybrid approach harnesses the advantages of both anomaly-based and signature-based detection methods. The techniques utilized include the stacking ensemble technique, which combines multiple machine learning approaches to improve the effectiveness of intrusion detection systems. The evaluation utilizes the Network Security Laboratory-Knowledge Discovery in Databases (NSL-KDD) and Australian Defense Force Academy (ADFA) datasets, recognized as benchmark datasets for intrusion detection. The proposed HIDS offers advantages such as high detection accuracy and low false alarm rates, effective for detecting both well-known intrusions and zero-day attacks. The system surpasses traditional IDS models by leveraging the combined strengths of SIDSs (signature-based intrusion detection systems) and AIDSs (anomaly-based intrusion detection systems). However, a notable drawback is the risk of high false alarm rates, particularly in dynamic and evolving cyber-attack environments. The authors propose future research directions that include refining the feature selection process and integrating advanced attack detection techniques to further enhance detection accuracy.

Shafin S.S. et al. [32] presents a deep learning-based detection system designed to identify obfuscated memory malware (OMM), specifically tailored for resource-constrained environments such as IoT devices. The system utilizes a hybrid architecture that integrates convolutional neural networks (CNNs) and bidirectional long short-term memory (Bi-LSTM) networks to manage the detection and classification of various types of obfuscated memory malware (OMM). The study developed two models, CompactCBL and RobustCBL, incorporating a structure consisting of a two-layer CNN followed by a two-layer Bi-LSTM block. A thorough tuning of model parameters was carried out to optimize performance while considering resource constraints. The models underwent evaluation using the CIC-Malmem-2022 dataset, which encompasses diverse malware types such as ransomware, spyware, and trojans. The proposed models demonstrate superior performance in terms of detection accuracy and efficiency compared to existing methods. They are lightweight and ideal for deployment on IoT devices, with the CompactCBL model especially notable for its compact size of 577 kB. The study highlights challenges in attaining high accuracy for detecting specific attack types within the OMM category. Future work will aim to improve detection accuracy for specific attack types and unknown (zero-day) attacks by employing semi-supervised and unsupervised learning models.

Ravi and Shalinie in [33] introduce an innovative security framework for intrusion detection and mitigation in IoT networks, utilizing a semi-supervised learning approach. This approach integrates labeled and unlabeled data to improve detection accuracy and enhance

model robustness. The framework uses feature selection techniques that combine Pearson correlation, rank-based chi-square, and score correlation to identify relevant features. It then applies an extreme gradient ensemble boosting method to categorize attack types. The framework underwent evaluation using the USNW-NB15, NSL-KDD, and CCIDS2017 datasets, recognized as standard benchmarks in intrusion detection systems. The proposed method achieved high accuracy rates: 99.96% for USNW-NB15, 97.48% for NSL-KDD, and 99.93% for CCIDS2017. Its minimal computational complexity and high detection rate make it well suited for real-time IoT security applications. Yet, challenges arise from the potential misclassification stemming from the varied and dynamic nature of IoT network traffic, compounded by the model's dependence on the quality and representativeness of labeled data. Future avenues for research encompass refining feature selection methodologies, incorporating advanced machine learning algorithms, and investigating blockchain applications for decentralized security enhancements within IoT networks.

Li, Meng, and Au, in [34], present an approach to enhance collaborative intrusion detection systems (CIDSs) in IoT environments through the application of semi-supervised learning. The study aims to overcome the issue of insufficient labeled data by harnessing unlabeled data. The DAS-CIDS system employs a semi-supervised learning algorithm based on disagreements. This method enables the system to leverage large quantities of unlabeled data to improve detection performance, minimizing the need for extensive human intervention in the labeling process. The authors assessed the effectiveness of DAS-CIDS through evaluations using simulated datasets and real-world IoT network environments, focusing on intrusion detection efficacy and false alarm reduction. The key benefit of DAS-CIDS lies in its automatic utilization of unlabeled data, thereby reducing the dependence on expensive and labor-intensive labeled data acquisition. As a result, it achieves enhanced detection performance and significantly lowers false alarm rates compared to conventional supervised classifiers. The study highlights concern regarding the algorithm's applicability across various IoT environments and underscores the necessity for additional testing to ensure its scalability and robustness. The authors propose investigating advanced semi-supervised learning methods and improving the system's ability to adapt to diverse IoT environments. Furthermore, future research should consider integrating alternative machine learning algorithms and enhancing the system's scalability.

In the work [35], Kwon et al. present a framework named Cyber Threat Dictionary (CTD), which maps attacks to defense mechanisms, aligning the MITRE ATT&CK matrix with the NIST framework. The CTD comprises a search engine offering attack details and solutions, along with a suggestion component for providing appropriate countermeasures.

The current studies offer promising approaches to IoT intrusion detection; however, they also demonstrate significant shortcomings. Yaras et al. [26] achieve high accuracy using CNN and LSTM models but encounter extended training durations attributed to large datasets. Harahsheh et al. [27] illustrate vulnerabilities in feature caching stability, affecting consistency when there are changes in dataset content. Javed et al. [28] demonstrate significant accuracy using random forest, yet their approach is constrained in its adaptability to various IoT environments. Liao et al. [29] exhibit superior performance through their ensemble framework, yet they face difficulties in handling the complexity and computational resources necessary for effectively implementing ensemble methods. Almaraz-Rivera et al. [30] face difficulties in achieving generalized self-supervised learning across new attack types. Ansam Khraisat et al. [31] achieve high accuracy but still experience significant false alarms in dynamic cyber-attack scenarios. Shafin S.S. et al. [32] encounter challenges in achieving consistent accuracy in detecting OMM and zero-day attacks.

Considering earlier investigations, crafting an efficient intrusion detection and prevention system (IDPS) requires tackling feature engineering for high-dimensional data and choosing potent detectors. In this study, we present a pioneering method employing autoencoders (AEs) and three distinct anomaly detectors, enhancing IDPS efficacy in identifying and mitigating potential security threats to network integrity.

3. Dataset

3.1. CSECIC-IDS2018

In 2018, the Canadian Institute of Cybersecurity (CIC) collaborated with the Communications Security Establishment (CSE) to develop the CSECIC-IDS2018 dataset [36]. Initially intended for evaluating intrusion detection research, it has evolved into a crucial benchmark for assessing IDSs. This carefully curated dataset mirrors real-world cyber threats and attacks, offering a range of scenarios for thorough analysis. Its importance stems from its ability to simulate intricate network environments, enabling both academics and practitioners to effectively assess and enhance intrusion detection systems. The dataset was gathered over a span of ten days, encompassing 80 columns, seven attack families, and fifteen distinct attack types. As shown in Table 1, it meets critical criteria such as overall traffic volume, a variety of attack types, and thorough labeling. The simulated attacks targeted an infrastructure consisting of 50 machines within an organization composed of five departments, 420 machines, and 30 servers. The dataset contains detailed records of network traffic and system logs for individual machines, utilizing CICFlowMeter-V3 to extract 80 attributes. The CSECIC-IDS2018 dataset comprises ten CSV files, encompassing 16,232,943 instances and 80 features. The distribution of attacks is shown in Figure 3.

Table 1. CSECIC-IDS2018 dataset network traffic.

Class	Type	Count	Total
Benign	Benign	-	13,484,708
DDoS	HOIC	686,012	1,263,933
	LOIC-UDP	1730	
	LOIC-HTTP	576,191	
DoS	Hulk	461,912	654,300
	GoldenEye	41,508	
	Slowloris	10,990	
	SlowHTTPTest	139,890	
Brute force	FTP	193,360	380,949
	SSH	187,589	
Bot	Bot	-	286,191
Infiltration	Infiltration	-	161,934
Web	Web	611	928
	XSS	230	
	SQL Injection	87	
Total instances			16,232,943

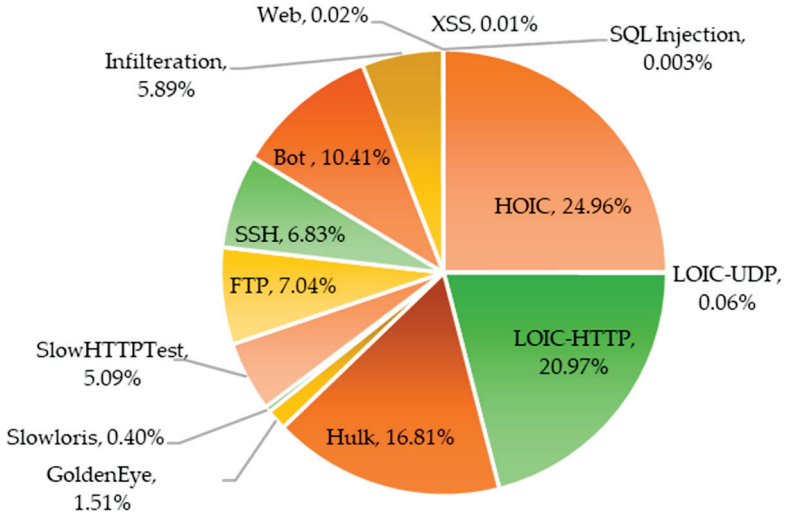


Figure 3. Distribution of attacks in CSECIC-IDS2018 dataset.

3.2. CIC-IDS2017

The CIC-IDS2017 dataset, created by the Canadian Institute of Cyber Security (CIC) [36], profiles intrusion traffic using a simulated attack scenario involving seven attack families.

The testbed infrastructure was partitioned into an attack network and a victim network, each equipped with standard devices such as firewalls, routers, switches, and a variety of operating systems (including Linux, Windows, and Macintosh). The dataset emphasizes a wide range of attacks, anonymity, inclusion of protocols, comprehensive traffic capture, network configuration details, metadata, labeled data samples, and diverse feature characteristics. In contrast to the restricted attack categories found in KDDCUP99 and NSL-KDD, the CIC-IDS2017 dataset encompasses a wider spectrum of attacks, including DDoS, DoS, brute force, XSS, SQL injection, botnet activities, web attacks, and infiltration attempts, as shown in Table 2. The dataset consists of 2,830,540 labeled flows, each containing 83 features categorized using the CICFlowMeter V4.0 tool, resulting in a dataset that is high-dimensional, multiclass, and imbalanced.

Table 2. CIC-IDS2017 dataset network traffic.

Instance Class	Number of Instances
Benign	2,359,087
DoS Hulk	231,072
PortScan	158,930
DDoS	41,835
DoS GoldenEye	10,293
FTP-Patator	7938
SSH-Patator	5897
DoS Slowloris	5796
DoS Slowhttptest	5499
Bot	1966
Web attack–brute force	1507
Web Attack–XSS	652
Infiltration	36
Web Attack–SQL injection	21
Heartbleed	11

4. Proposed Method for Intrusion Detection and Prevention

The proposed model for intrusion detection and prevention (IDP), which integrates bAE + OCSVM + dAE + DBSCAN + ATT&CK, referred to as hybrid IDPS, has been meticulously designed with a comprehensive structure encompassing key phases in the realm of cybersecurity. The model's framework includes traffic data preprocessing, feature extraction, training the detection model, conducting anomaly detection, and clustering and preventing attacks. This model adopts a hybrid approach, integrating a deep learning algorithm that incorporates various techniques for enhanced cybersecurity measures.

In the dimensionality reduction phase, the basic autoencoder (bAE) is employed to effectively reduce the number of dimensions. A one-class Support vector machine (OCSVM) is utilized for detecting anomalies, providing a robust mechanism during the training phase of the model. Additionally, the deep autoencoder (dAE) is employed for identifying abnormal behavior, with a specific focus on minimizing false negatives. The root mean square propagation (RMSP) algorithm serves as the optimization technique, contributing to the efficiency of the overall model.

The model's capacity for handling anomaly attacks is strengthened through the application of the DBSCAN algorithm during the clustering phase. Moreover, prevention strategies are integrated into the model, drawing on the insights and methodologies provided by the MITRE ATT&CK framework.

The architectural diagram of the proposed model, in Figure 4 further illustrates the intricate structure that integrates various techniques for a robust cybersecurity solution.

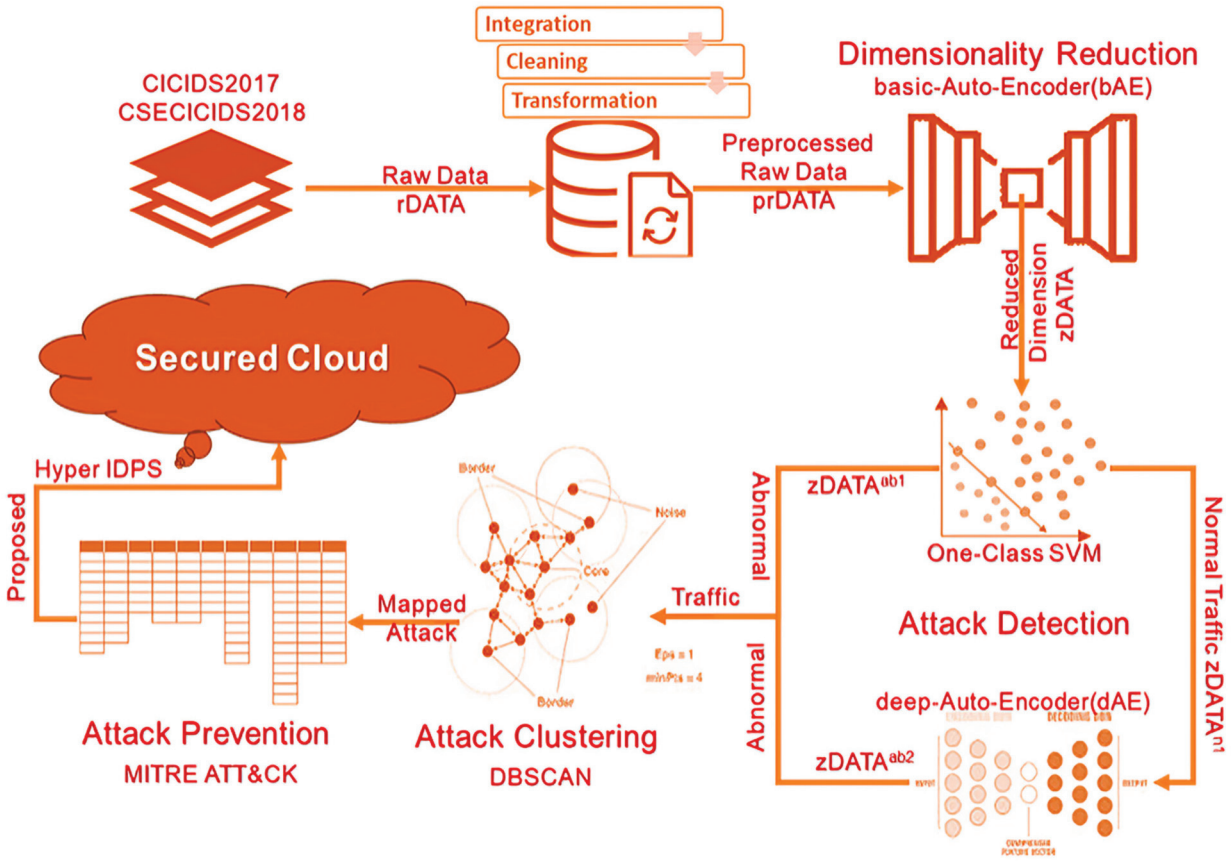


Figure 4. Architecture of the proposed model.

4.1. Data Preparation

The data preparation module has concentrated on three key segments:

1. Collecting data;
2. Preprocessing data;
3. Reducing dimensionality.

4.1.1. Data Collection

Data collection for intrusion detection systems (IDSs) involves consolidating raw data from diverse sources, including network traffic and system logs. Integration of incident information and external threat feeds enhances IDS capabilities. Despite challenges in real-world adoption, systematic dataset generation addresses testing limitations. Datasets like DARPA, KDD Cup 1999, NSL-KDD, ISCX IDS 2012, UNSW-NB15, CIC-IDS2017, and CSECIC-IDS2018 support research, facilitating intrusion detection system evaluation. In this research, raw data "rDATA" is derived from the CIC-IDS2017 and CSECIC-IDS2018 datasets, chosen for their contemporaneity and representation of real-world data.

4.1.2. Data Preprocessing

Data preprocessing is crucial post-collection, especially in machine learning. ISCX offers datasets like CIC-IDS2017 and CSECIC-IDS2018 in CSV format, containing benign and attack traffic. Normalization is vital due to undistributed histogram data, ensuring uniform scale via Z-score standardization. Each data point's Z-score (Zi) is calculated using the mean (μ) and standard deviation (σ) of the dataset as expressed by the formula in Equation (1):

$$Zi = \frac{(x - \mu)}{\sigma} \quad (1)$$

4.1.3. Dimensionality Reduction

The preprocessed data, labeled as "prDATA" and comprising seventy-six features, undergoes dimensionality reduction through the basic autoencoder (bAE) as depicted in Figure 5. As a feature extraction technique, bAE systematically reduces dimensionality while considering all features' influence on the outcome. This process involves three internal submodules: the encoder, decoder, and loss function. Together, they contribute to extracting essential features, resulting in a condensed feature representation.

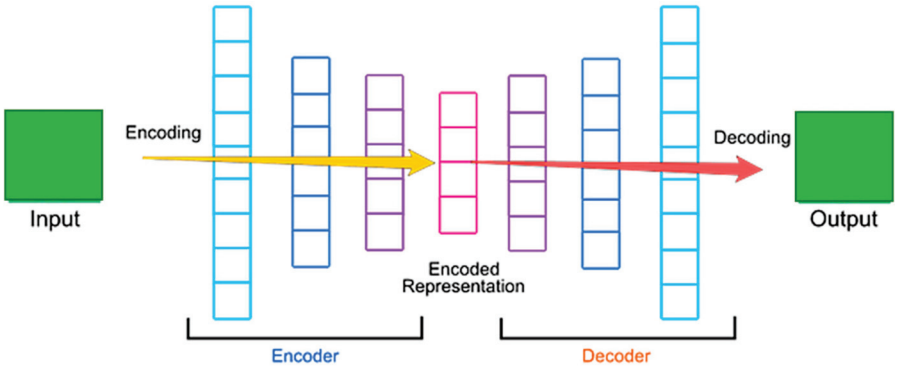


Figure 5. The architecture of the proposed autoencoder.

The encoder transforms "prDATA" into a lower-dimensional "zDATA" using weights W and biases b with an activation function f. Among activation functions, the rectified linear unit (ReLU) is represented as max(0, prDATA), while the sigmoid activation function is defined as in Equation (2):

$$Sigmoid = \frac{1}{(1 + \acute{e}(-prDATA))} \tag{2}$$

ReLU and sigmoid introduce non-linearity into the model's output. The decoder aims to reconstruct the original input data "rcDATA" from "zDATA" using weights W and biases b with another activation function "g". The loss function, typically the mean squared error (MSE), quantifies the disparity between "prDATA" and "rcDATA". The reconstruction formula is defined in Equation (3) as follows:

$$rcDATA = g\left(\left(W' * zDATA\right) + b'\right) \tag{3}$$

After dimensionality reduction, the features, now in "zDATA", are ready for further processing. These reduced-dimensional data points, derived from the latent space, are crucial for subsequent analytical and modeling tasks.

4.2. Detection Stage 1: OCSVM

One-class support vector machine (OCSVM) is a distinctive method in one-class classification, focusing solely on a singular data class, unlike binary or multiclass methods [12]. After training, the model assesses whether a new sample belongs to the designated target class. OCSVM operates in a high-dimensional kernel space F by transforming samples from "zDATA" using a mapping function ϕ. The inner product within F is computed using a kernel function k aiming to find a hyperplane that maximizes the separation between data and the origin. OCSVM addresses a quadratic problem, formulating a hyperplane with a weight vector w and margin ρ, allowing for soft margins via non-negative slack variables ε. The hyperparameter ν controls the trade-off, acting as an upper limit on the proportion of anomalies. In the OCSVM framework, w signifies the weight vector of the hyperplane, and ρ represents the margin. The inclusion of non-negative slack variables ε introduces flexibility by permitting some samples to extend beyond the hyperplane, softening the margin. The hyperparameter ν, constrained within (0, 1), plays a crucial role in

balancing the trade-off and acts as an upper limit on the proportion of anomalies, affecting the model's sensitivity to outliers. The Gaussian kernel function, as employed in Equation (4), effectively represents data in the kernel space, with the hyperparameter γ controlling the shape of the decision boundary in the feature space. Following problem resolution, the decision function f(x) is calculated for a given testing sample as in Equation (5):

$$k(x, y) = exp\left(-\gamma \parallel x - y \parallel^2\right) \tag{4}$$

$$f(x) = Sgn(< w, \theta(x) > f - \rho) \tag{5}$$

Anomalies are identified, as illustrated in Figure 6, when new samples fall on the incorrect side of the hyperplane, resulting in negative values in the decision function for anomalous points, denoted as "zDATA^{ab1}". Conversely, normal samples positioned correctly on the hyperplane yield positive values in the decision function, denoted as "zDATA^{n1}".

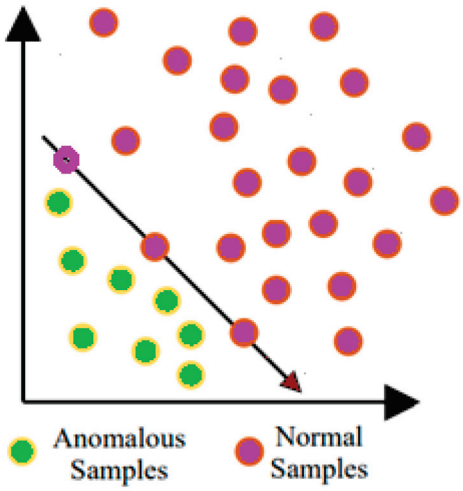


Figure 6. One-class support vector machine illustration.

4.3. Detection Stage 2: OCSVM + Deep Autoencoder (dAE)

The deep autoencoder (dAE) is pivotal in the second stage of our detection process. Its design focuses on operating as a deep learning classifier, adept at handling lower-dimensional data, especially in clustered data scenarios. This section outlines how the dAE leverages its capabilities to effectively differentiate between normal and anomalous traffic.

4.3.1. Input Data and Classification

In the context of normal traffic, represented by zDATA^{n1}, the output from the OCSVM classifies normal traffic, supplying essential data for the dAE neural network, and the dAE emerges as a deep learning classifier. The dAE is trained primarily on non-attack packet data, improving its capacity to identify normal patterns and behaviors effectively.

4.3.2. Deep Learning Classifier

It thrives on processing lower-dimensional data, which showcases optimal performance, especially in clustered data scenarios. Its prowess lies in capturing meaningful patterns and behaviors, making it adept at discerning normal observations. Focused solely on non-attack packet data, the dAE stands out in this domain by effectively classifying attacks through its learned knowledge from the training dataset.

4.3.3. Training Process

Operating through the iterative process of backpropagation, the deep autoencoder leverages training data results to learn, involving both decoder and forward propagation

phases. These phases constitute stages of the encoder, visually depicted in Figure 5. With a Z-dimensional vector, the encoder function (e) is defined in Equation (6). Encoded data undergo processing, reverse-propagating for decoding using the original data. The decoder function (d) is parameterized as shown in Equation (7), enabling multiple encode and decode processes on hidden layers. During training, the objective is to minimize the mean squared error between the input data and the reconstructed output. This ensures that the model effectively captures the key features and patterns of normal traffic. This reconstruction error indicates the model's ability to accurately reconstruct normal data.

$$E_i = e\left(zDATA^{n1}{}_i, \theta_e\right) \tag{6}$$

$$D_i = d(E_i, \theta_d) \tag{7}$$

4.3.4. Backpropagation and Cost Minimization

Backpropagation on encoded data occurs using a mean squared error cost minimizer. This process is utilized on test data as well. The goal is to reduce the reconstruction error, thus enhancing the model's accuracy. The comprehensive procedure can be outlined by integrating both the encoder and decoder functions, as depicted in Equation (8):

$$D_i = d\left(e\left(zDATA^{n1}{}_i, \theta_e\right), \theta_d\right) = g\left(zDATA^{n1}{}_i, \theta\right) \tag{8}$$

4.3.5. Establishing the Threshold

After completing the training process, the distribution of reconstruction errors pertaining to the normal data is analyzed. Typically, this distribution reveals a clustering of low reconstruction errors, demonstrating the model's effectiveness in reconstructing normal traffic. To determine the anomaly detection threshold, the standard deviation method is used on the reconstruction errors.

4.3.6. Output and Classification

During the detection phase, the dAE handles new incoming data, which could comprise both normal and anomalous samples. When a new instance is input to the dAE, it undergoes encoding followed by decoding using the learned functions e and d. If the instance is normal, the reconstruction error will be minimal, reflecting the model's familiarity with normal data pattern. In contrast, anomalous data points will not align well with the learned patterns of the dAE. When an anomalous sample undergoes encoding and decoding, the reconstruction error will be notably higher due to the model's lack of training on these patterns. The elevated reconstruction error serves as a signal or indicator of an anomaly occurring. The data points characterized by high reconstruction errors are identified as abnormal and categorized under "$zDATA^{ab2}$".

The output "$zDATA^{ab2}$" from the dAE algorithm represents the classified attack data. The algorithm efficiently reduces the mean squared error, thereby improving its capability to accurately classify normal and anomalous traffic.

4.4. Detection Stage 3: OCSVM + dAE + DBSCAN Clustering

In this research, DBSCAN is utilized due to its robustness in handling noise and its capability to identify clusters of arbitrary shapes without needing a predefined number of clusters. Moreover, its flexibility in parameter tuning and scalability are essential for enhancing the effectiveness of intrusion detection. This section explains how DBSCAN applies these capabilities to effectively cluster both normal traffic and various attack anomalies.

The outputs "$zDATA^{ab1}$" from OCSVM and "$zDATA^{ab2}$" from stage 2 dAE, respectively, contain information about attack patterns, representing points within the output space. DBSCAN operates on these data using the ε (epsilon) and MinPts parameters, initiating its clustering process from an unvisited point. The ε parameter defines the maximum distance for considering neighborhood points, essentially setting a radius for each

point. DBSCAN identifies clusters by retrieving ε-neighborhoods, forming clusters if they are sufficiently populated, otherwise classifying as noise. MinPts signifies the minimum number of samples required for a point to be considered a core point, which serves as the central element within dense regions. The fit method assigns clusters based on density, expanding to densely populated ε-neighborhoods. This iterative process continues until the entire density-connected cluster is identified. New unvisited points are sequentially processed, potentially revealing additional clusters or noise. DBSCAN optimizes clustering by minimizing the number of clusters from all possible clustering within the dataset. It ensures that each pair of points within a cluster is density-reachable, maintaining the original properties of clusters in terms of "maximality" and "connectivity".

4.5. Prevention: MITRE ATT&CK

Utilizing the MITRE ATT&CK framework, we establish a comprehensive database, referred to as the "Cyber Threat Repository", cataloging various attacks alongside associated tactics, assigned priorities, and corresponding mitigation strategies. As shown in Figure 7, upon detecting a potential attack, if it matches an entry within the repository, its priority level is promptly identified for subsequent actions. Attacks deemed high-priority trigger an immediate response, where the identified data traffic is swiftly intercepted and blocked by the "Security Enforcement Gateway", while simultaneous notifications are dispatched to relevant stakeholders. Conversely, low-priority incidents prompt the redirection of associated data traffic to a designated repository known as the "Incident Log Database", facilitating subsequent analysis and prompting alerts to cybersecurity personnel for targeted mitigation measures in accordance with the repository's guidance.

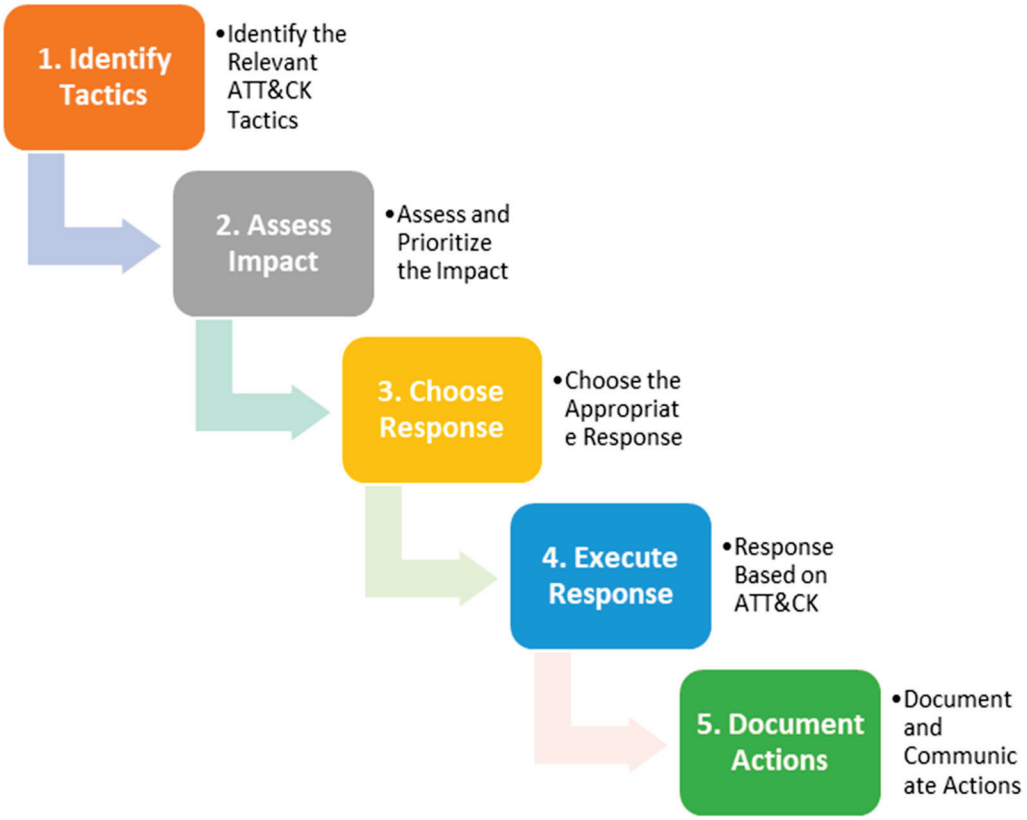


Figure 7. MITRE ATT&CK response process.

The suggested model can be implemented by placing it between the firewall and the network router. This allows for it to predict potential attacks within the network after being trained on actual data sourced from the network. The data are collected, prepared, and

utilized for training the model. The trained model has the capability to analyze the data flowing through the network in real time.

5. Results

The suggested approach undergoes evaluation on the seven attacks within each dataset as shown in Table 3, utilizing the CSECIC-IDS2018 and CIC-IDS2017 [36] intrusion detection datasets. The evaluation entails contrasting the proposed model with standard clustering techniques (OPTICS [37], fuzzy C-means [38], K-means [39], and hierarchical [40]) using un-clustered data, followed by a performance assessment.

Table 3. Experimental samples.

	CIC-IDS2017	**CSECIC-IDS2018**
Training class	Benign, DoS, DDoS, botnet and brute force	Benign, DoS, DDoS, botnet and brute force
Testing class	Training class + Heartbleed and infiltration	Training class + web attack and infiltration
Numbers of features	83	80
Number of benign instances	2,359,087	2,374,871
Number of attack instances	224,893	239,842

5.1. Experimental Dataset Usage

Table 3 displays the dataset utilized and the components of selected samples for this study, while Table 4 shows the configuration of hyperparameters. Performance metrics are evaluated to gauge the effectiveness of the intrusion detection model, utilizing a confusion matrix. This assessment includes accuracy, precision, recall, specificity, false negative rate, and F-score.

Table 4. Hyperparameter configuration.

Hyperparameter	**Values**
nu(v)	0.095
Kernel	Gaussian
Optimizer	Adam
Learning rate	0.001
Epochs	40
Batch size	32
Patience	8
Latent space	3
Threshold	0.0219
Epsilon (ε)	0.2
MinPts	800

Precision, as defined in Equation (9), measures the accuracy of optimistic predictions. Recall, represented by Equation (10), captures the ability of the model to correctly classify positive instances. Specificity, outlined in Equation (11), provides valuable insights into the model's effectiveness in identifying instances of the negative class accurately. The F-score, presented in Equation (12), offers a comprehensive evaluation metric that balances precision and recall. Accuracy, as indicated in Equation (13), quantifies the proportion of correct classifications made by the model. The false negative rate (FNR), or miss rate, quantifies the percentage of actual positive instances incorrectly identified as negatives by the model, as defined in Equation (14).

In evaluating the efficacy of the prevention model, key performance indicators including prevention rate, priority-based blocking rate, and success rate are employed. The prevention rate, as depicted in Equation (15), signifies the proportion of thwarted threats. The priority-based blocking rate, illustrated in Equation (16), focuses on intercepting high-priority threats. Equation (17) measures the overall effectiveness of the model. These

metrics offer valuable insights into the model's ability to mitigate threats and prioritize actions, crucial for effective cybersecurity management and risk mitigation strategies:

$$Precision = \frac{True\ Positive}{Predicted\ Positives} \tag{9}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{10}$$

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive} \tag{11}$$

$$F - Measure = \frac{2 * (Precision * Recall)}{Precision + Recall} \tag{12}$$

$$Accuracy = \frac{True\ Positive + True\ Negative}{Predicted\ Positive + Predicted\ Negative} \tag{13}$$

$$False\ Negative\ Rate = \frac{False\ Negative}{True\ Positive + False\ Negative} \tag{14}$$

$$Prevention\ Rate = \frac{Number\ of\ detected\ attacks\ mitigated}{Total\ number\ of\ detected\ attacks} \tag{15}$$

$$Priority\ based\ blocking\ Rate = \frac{Number\ of\ High\ priority\ attacks\ Blocked}{Total\ number\ of\ High\ priority\ attacks\ detected} \tag{16}$$

$$Success\ Rate = \frac{Number\ of\ attacks\ prevented}{Total\ number\ of\ detected\ attacks} \tag{17}$$

The effective performance of the model hinges on critical elements like the suitable architecture and hyperparameter configurations of the detection model. A series of comprehensive experiments were carried out to identify the most effective hyperparameter configurations. The bAE in the proposed method is composed of a solitary input layer and a sole output layer. The OCSVM model is designed with a single class, while the dAE consists of three latent space. These layers are structured with unit sizes tailored to align with the loss function, ensuring an optimal fit with the varying dimensions of features present within the training data.

5.2. Training and Testing of Detection Model

In this methodology, the dataset is split into training (70%) and testing (30%) subsets. The model training includes the utilization of a Gaussian kernel function to ensure effective representation of data in the kernel space for OCSVM. The initial value of the "nu" parameter (v) is set to 0.5 and adjusted through cross-validation, resulting in "nu" being set to 0.95. Early stopping criteria are implemented in dAE to determine the optimal epoch count. Additionally, ReLU serves as the activation function, while Adam optimization with a learning rate of 0.001 aids in model optimization during training.

5.3. Evaluation on Stage 1: OCSVM

Based on the results of detection stage 1 using OCSVM, we analyze the metrics depicted in confusion matrix Figure 8 for CSECIC-IDS2018 and Figure 9 for CIC-IDS2017. The model calculates the performance metrics as illustrated in Figure 10. The significant count of true negatives (2,316,485) demonstrates the model's effectiveness in identifying normal traffic accurately as non-attacks, highlighting its strong capability in distinguishing non-malicious activities. The false negatives (12,624) indicate cases where the model failed to detect actual attacks. While relatively low compared to true positives and true negatives, this number remains critical as missed detections could lead to undetected attacks.

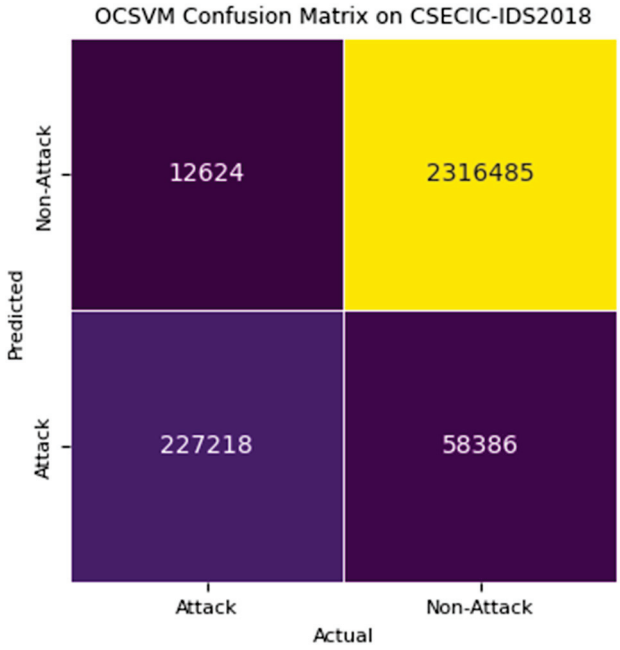


Figure 8. Confusion matrix of detection stage 1 OCSVM on CSECIC-IDS2018.

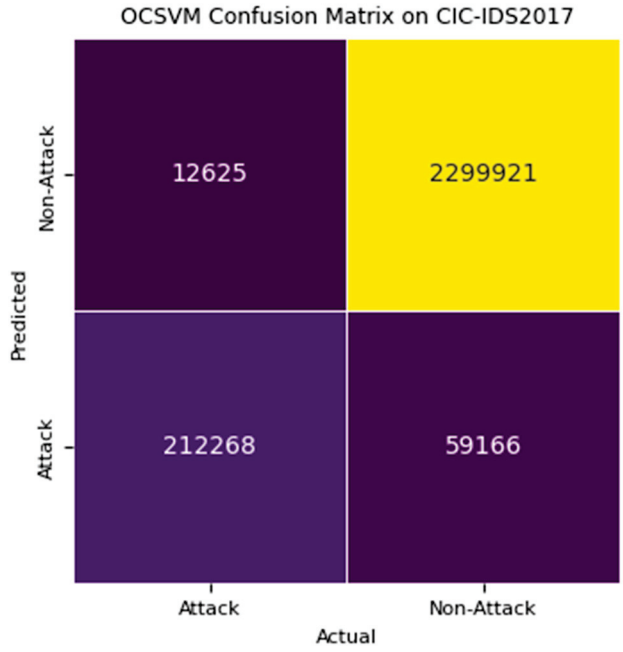


Figure 9. Confusion matrix of detection stage 1 OCSVM on CIC-IDS2017.

The model's high number of true positives (212,268) demonstrates its capability to accurately identify a substantial portion of attacks. This positive aspect highlights the model's reliability in identifying malicious activities, thereby contributing significantly to its high recall rate. The false positives (59,166) represent cases where normal traffic was mistakenly classified as an attack. This is problematic because false positives can result in unnecessary alerts, leading to potential alarm fatigue and challenges in resource allocation.

The higher incidence of false positives suggests that although the model is adept at detecting attacks, it tends to be overly cautious, labeling benign traffic as malicious. The assessment of detection accuracy in OCSVM relies on the "nu" parameter, which regulates the balance between model complexity (flexibility) and the acceptance of margin violations. The current "nu" parameter (set to 0.95) in the OCSVM model creates a decision boundary

that is overly stringent, leading normal traffic patterns to be misclassified as attack patterns. To optimize performance, we adjusted the "nu" parameter to 0.1, thereby relaxing the model's strictness and enhancing precision and specificity. Although the precision is moderately high, there is still room for improvement to meet elevated standards. However, this adjustment does affect other metrics. To mitigate false negatives overlooked by the OCSVM model and decrease the false positive rate, we implemented a hybrid approach incorporating a deep autoencoder (dAE) in the subsequent detection stage. The dAE model effectively identifies normal patterns, thereby enhancing overall accuracy, recall, and specificity.

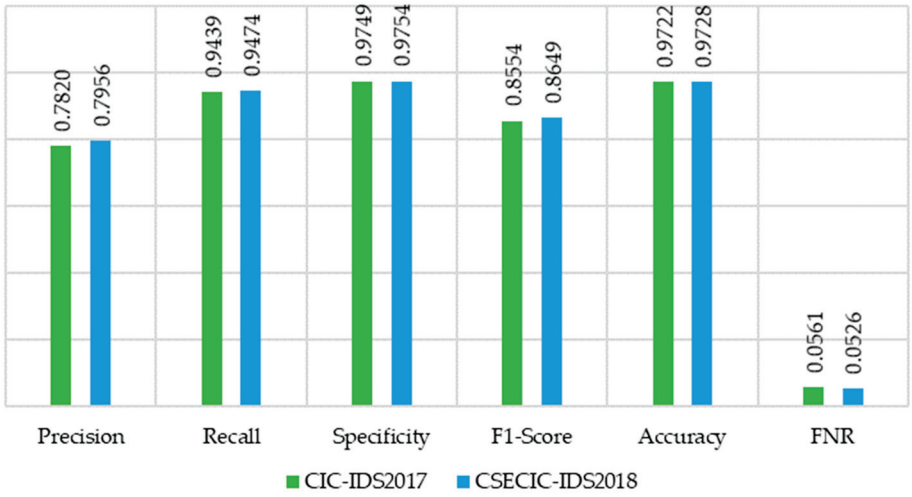


Figure 10. Performance measures of stage 1 detection OCSVM.

5.4. Evaluation on Stage 2: OCSVM + Deep Autoencoder (dAE)

The deep autoencoder (dAE) model operates as the second stage in the intrusion detection system, handling data flagged as negative by the stage 1 one-class SVM (OCSVM). The evaluation of detection stage 2, dAE, utilizes mean squared error (MSE) as the loss function, with a threshold value established at 0.0219. Figure 11 illustrates the stage 2 dAE detection model's loss function, where this research utilized a three-latent-space dAE, 40 epochs, 32 batches, and the Adam optimizer with a learning rate of 0.001 and patience of 8. The stage 2 dAE model's confusion matrix is shown in Figures 12 and 13 and its performance is depicted in Figure 14 across the CIC-IDS2017 and CSECIC-IDS2018 datasets, showcasing the effectiveness of the model. The detection model, constructed using both "Normal" and "Attack" data from these datasets, demonstrates significant performance across both datasets. The experimental results highlight its notable performance, which is comparatively higher than that of the stage 1 OCSVM.

The dAE demonstrates improved performance metrics across precision, recall, specificity, F1-score, and accuracy for both the CIC-IDS2017 and CSECIC-IDS2018 datasets, as depicted in Figure 14. Moreover, it notably reduces false positives, enhancing its ability to differentiate between normal and attack traffic beyond the initial screening by OCSVM. With a reduced false negative rate, the dAE enhances overall detection capabilities, highlighting its role in improving the reliability and efficiency of intrusion detection systems.

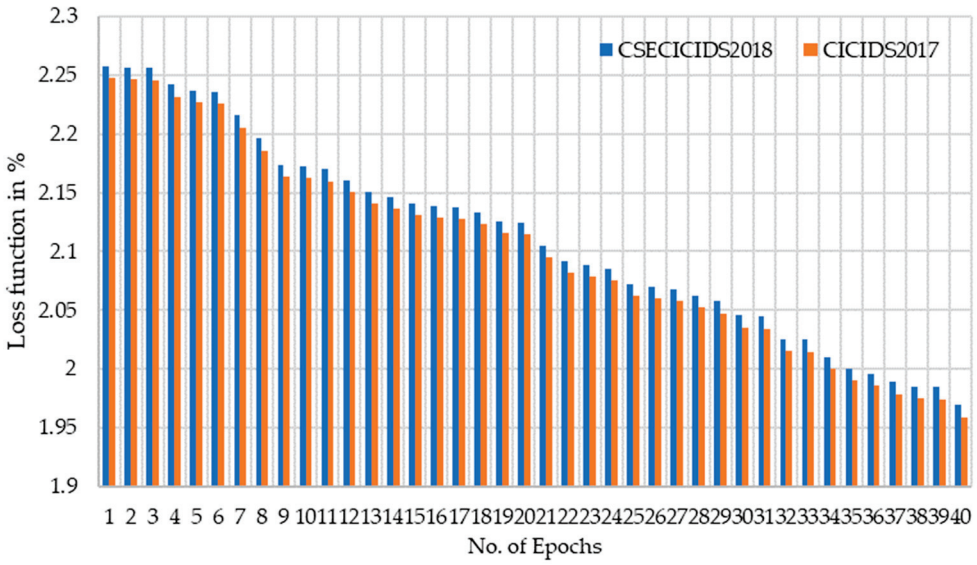


Figure 11. Detection stage 2 OCSVM + dAE model loss function.

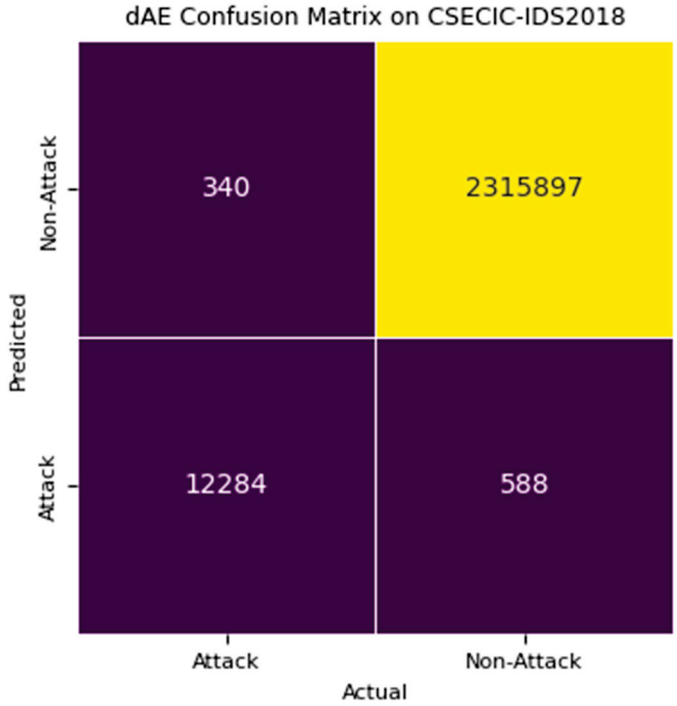


Figure 12. Confusion matrix of detection stage 2 OCSVM + dAE on CSECIC-IDS2018.

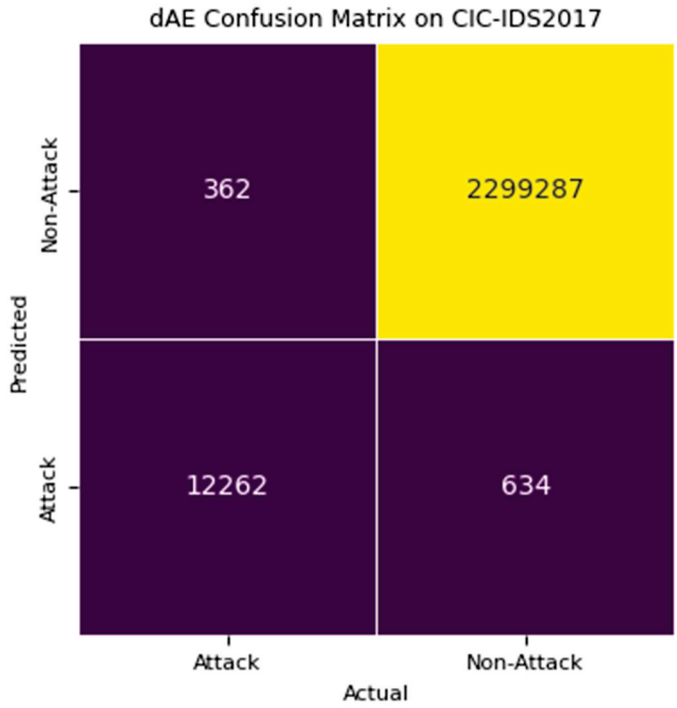


Figure 13. Confusion matrix of detection stage 2 OCSVM + dAE on CIC-IDS2017.

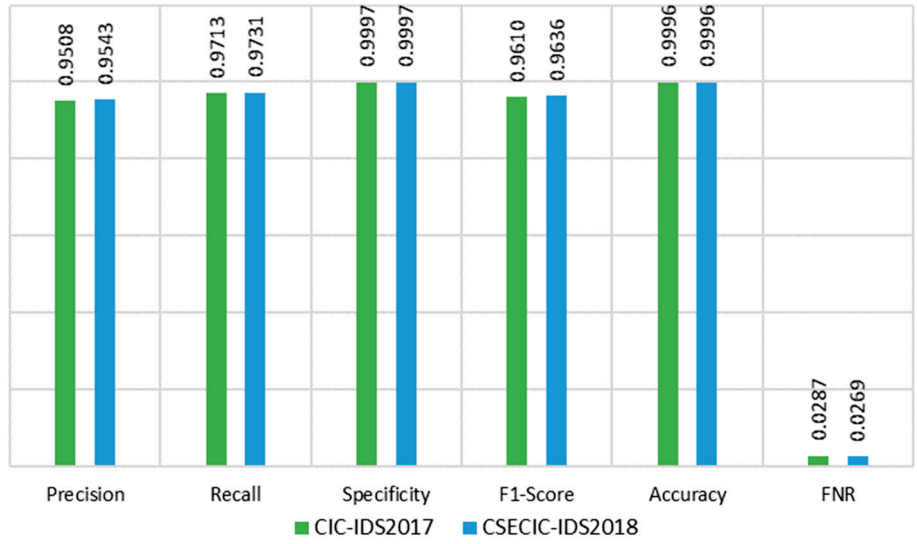


Figure 14. Performance measures of stage 2 detection OCSVM + dAE.

5.5. Evaluation on Stage 3: OCSVM + dAE + DBSCAN

The DBSCAN model operates as the third stage in the intrusion detection system, handling data flagged as positive by the stage 1 one-class SVM (OCSVM) and stage 2 dAE. The evaluation of the stage 3 DBSCAN uses an epsilon value of 0.2 and a minimum of 800 points for clustering. The clustering model is assessed against commonly used clustering algorithms including K-means [39], fuzzy C-means (FC-Means) [38], hierarchical clustering [40] (specifically divisive hierarchical clustering—DHC), and OPTICS [37], utilizing network traffic datasets for evaluation. The scatter plots in Figures 15 and 16 display the experimental results for CSECIC-IDS2018 and CIC-IDS2017, respectively. The experimental results show that the proposed clustering method effectively separates instances into eight groups, including benign ones, for both CSECIC-IDS2018 and CIC-IDS2017

datasets. It demonstrates high accuracy in identifying positive and negative instances while maintaining a balanced performance in classification.

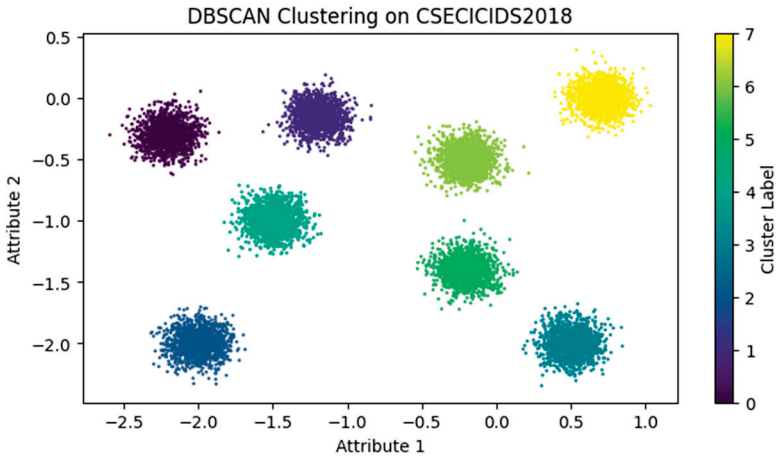


Figure 15. DBSCAN clustering on CSECIC-IDS2018.

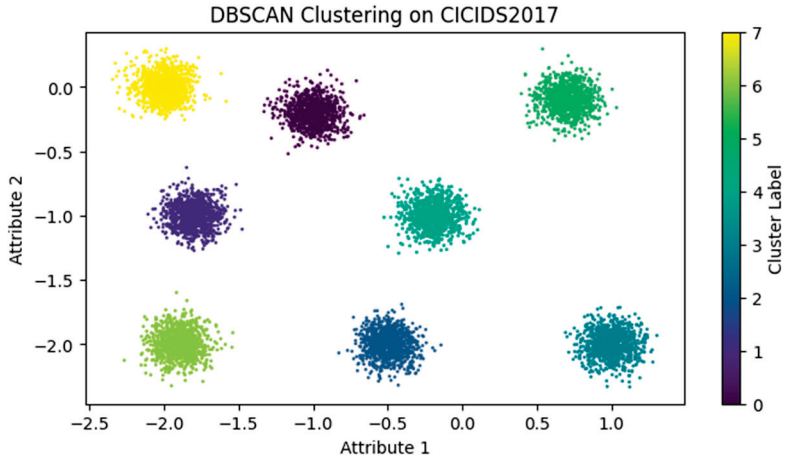


Figure 16. DBSCAN clustering on CIC-IDS2017.

Table 5 illustrates how the proposed method consistently outperforms alternative clustering algorithms across a range of metrics, highlighting its effectiveness in clustering tasks in the CIC-IDS2017 dataset. The proposed method attains the highest recall score (0.9907), indicating its robust capability to identify positive instances effectively, also suggesting a minimal false negative rate.

Table 5. Performance comparison of stage-3 clustering OCSVM + dAE + DBSCAN on CIC-IDS2017.

Methods	Precision	Recall	Specificity	Accuracy	F-Measure	FNR
OPTICS [37]	0.9331	0.9345	0.9298	0.9324	0.9338	0.0655
FC-Means [38]	0.9451	0.9464	0.9418	0.9443	0.9458	0.0536
K-Means [39]	0.9540	0.9553	0.9507	0.9532	0.9547	0.0447
Hierarchical [40]	0.9509	0.9522	0.9476	0.9501	0.9516	0.0478
Proposed	0.9948	0.9907	0.9806	0.9927	0.9886	0.0093

Figure 17 illustrates that the proposed method outperforms others in the CSECIC-IDS2018 dataset, showcasing higher F1-score, recall, precision, FNR, and accuracy. The proposed approach achieves the highest overall correctness of clustering at 0.9881. The proposed approach achieves the highest F1-measure at 0.993, signifying a well-balanced performance between precision and recall.

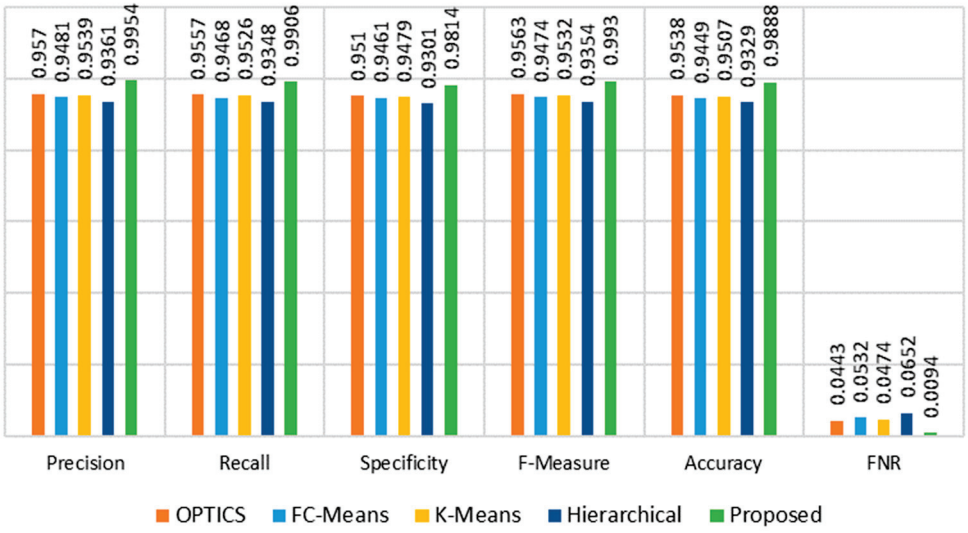


Figure 17. Performance comparison of stage 3 clustering OCSVM + dAE + DBSCAN on CSECIC-IDS2018.

5.6. Performance Analysis of Proposed Method

5.6.1. Analysis on CSECIC-IDS2018 Dataset

The CSECIC-IDS2018 and CIC-IDS 2017 datasets were assessed using various anomaly detection methods. Key metrics such as precision, recall, specificity, accuracy, F-measure, and false negative rate (FNR) were used in the evaluation. Table 6 presents the analysis of these metrics for different methods applied to the CSECIC-IDS2018 dataset. This is followed by a detailed explanation of our proposed intrusion detection approach, which involves OCSVM + dAE + DBSCAN.

Table 6. Performance analysis of proposed IDS on CSECIC-IDS2018 dataset.

Methods	Precision	Recall	Specificity	F-Measure	Accuracy	FNR
					Detection Stage 1:	
OCSVM	0.7956	0.9474	0.9754	0.8649	*0.9728*	*0.0526*
dAE	0.8008	0.9453	0.9763	0.8671	0.9734	0.0547
DBSCAN	0.7916	0.9328	0.9752	0.8564	0.9713	0.0672
					Detection Stage 2:	
OCSVM + DBSCAN	0.9561	0.9526	0.8298	0.9544	0.9275	0.0474
dAE + DBSCAN	0.9556	0.9525	0.8222	0.9541	0.9266	0.0475
dAE + OCSVM	0.7772	0.9536	0.9985	0.8564	0.9982	0.0464
OCSVM + dAE	*0.9543*	*0.9731*	*0.9997*	*0.9636*	*0.9996*	*0.0269*
					Detection Stage 3:	
dAE + OCSVM + DBSCAN	0.9504	0.9697	0.7983	0.9600	0.9353	0.0303
OCSVM + dAE + DBSCAN	*0.9954*	*0.9906*	*0.9814*	*0.9930*	*0.9888*	*0.0094*

Stage 1 Methods

OCSVM demonstrates high recall and specificity, leading to a low false negative rate (FNR), indicating slightly superior performance compared to dAE and DBSCAN. The dAE method achieves good recall and specificity, albeit with a slightly higher false negative rate (FNR). In contrast, DBSCAN exhibits lower recall and specificity than OCSVM and dAE, resulting in a higher false negative rate (FNR). Among the three methods, OCSVM performs exceptionally well, particularly in recall and achieving a low false negative rate, demonstrating strong capability in detecting actual anomalies. It is thus considered a recommended method for stage 1 detection.

Stage 2 Combined Methods

The rationale for implementing a two-stage anomaly detection approach with OCSVM and dAE is to leverage the respective strengths of each technique, aiming for improved detection performance. OCSVM and dAE exhibit complementary performance both individually and together.

OCSVM correctly identified 227,218 as true positive samples, while dAE detected 226,719, indicating that OCSVM found an additional 499 samples missed by dAE. On the other hand, dAE identified 226,719 samples correctly, while OCSVM detected 227,218, showing that dAE missed 499 samples that OCSVM detected. The combined use of OCSVM and dAE effectively reduces false negatives and false positives to just 340 and 588, respectively. This underscores the method's effectiveness in minimizing detection errors.

The pairing of OCSVM and DBSCAN achieves high precision and recall but shows decreased specificity, characterized by a lower false negative rate (FNR). The combination of dAE and DBSCAN shows comparable performance, marked by notable enhancements in precision and recall. The combination of dAE and OCSVM achieves high recall but at the cost of relatively lower precision. The combined method of OCSVM and dAE shows substantial improvements in precision, recall, and exceptional specificity, resulting in a notably low false negative rate (FNR). This OCSVM + dAE approach is proposed for stage 2 consideration.

The two-stage approach improves anomaly detection by leveraging the strengths of both OCSVM and dAE, resulting in a more robust and accurate detection mechanism. The combined method significantly decreases false negatives and false positives compared to using each method individually, providing a strong rationale for adopting a two-stage anomaly detection strategy.

Stage 3 Proposed Method

The combination of dAE, OCSVM, and DBSCAN demonstrates strong overall performance with high precision and recall, albeit with reduced specificity. The proposed method, integrating OCSVM, dAE, and DBSCAN, shows superior performance across all metrics—precision, recall, specificity, accuracy, and notably low false negative rate (FNR), as shown in Table 6.

5.6.2. Analysis on CIC-IDS2017 Dataset

The CIC-IDS2017 dataset underwent evaluation using three distinct anomaly detection methods: OCSVM, dAE, and DBSCAN. The performance metrics assessed include precision, recall, specificity, F1-score, accuracy, and false negative rate (FNR).

Stage 1 Method

As shown Figure 18, among the three methods, DBSCAN exhibits the lowest precision and recall, suggesting a higher number of false positives and missed anomalies. Its specificity is marginally lower compared to OCSVM and dAE, and its F1-score indicates the least balance between precision and recall. DBSCAN also demonstrates the lowest accuracy and the highest false negative rate. The dAE method shows marginally superior precision and recall compared to the other methods. Despite its slightly lower recall, it maintains a competitive false negative rate, demonstrating its effectiveness in distinguishing between anomalies and normal instances.

This proposed OCSVM demonstrates a precision of 0.7820, indicating that 78.2% of its positive predictions were correct, and a recall of 0.9439, successfully identifying 94.39% of the actual anomalies. Its specificity is 0.9749, showing that it correctly identified 97.49% of normal instances, and it achieves an F1-score of 0.8554, demonstrating a balanced trade-off between precision and recall. OCSVM demonstrates an accuracy rate of 97.22%, indicating that nearly 97.22% of predictions were correct. It also exhibits a false negative rate (FNR) of 0.0561, implying that approximately 5.61% of true anomalies were not detected. Among the three methods, OCSVM stands out with notably better recall and a lower false negative rate compared to the others. This highlights its strong capability in effectively identifying true anomalies.

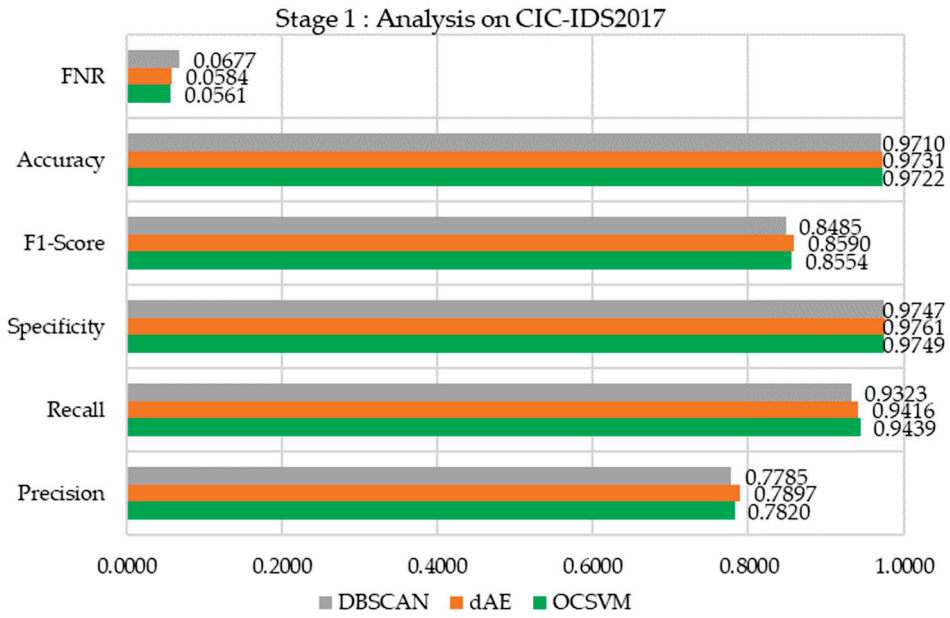


Figure 18. Performance analysis of stage 1 OCSVM on CIC-IDS2017.

Stage 2 Method

As shown in Figure 19, the combination of OCSVM + DBSCAN demonstrates high precision and recall, effectively identifying true positives while minimizing false positives. Its lower specificity suggests some difficulties in accurately identifying normal instances. The F1-score demonstrates a well-balanced performance between precision and recall, achieving solid accuracy alongside a low false negative rate (FNR).

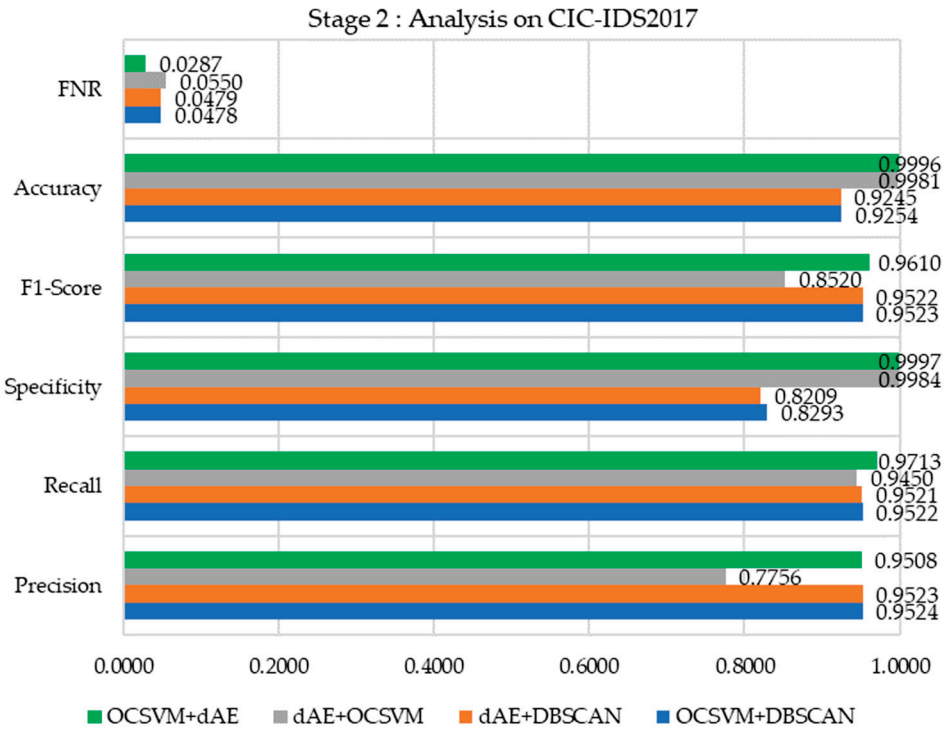


Figure 19. Performance analysis of stage 2 OCSVM + dAE on CIC-IDS2017.

Like OCSVM + DBSCAN, the method dAE + DBSCAN exhibits high precision and recall. However, its slightly lower specificity suggests some challenges in accurately identifying normal instances. The dAE + OCSVM combination demonstrates strong performance

in achieving high recall and excellent specificity, effectively identifying anomalies and normal instances. However, its precision is comparatively lower, leading to a higher incidence of false positives.

The OCSVM + dAE combination proposed here demonstrates superior performance across most metrics, characterized by high precision, recall, and nearly flawless specificity. The F1-score and accuracy excel, accompanied by the lowest false negative rate (FNR) among the methods, underscore its exceptional ability to detect anomalies while minimizing both false positives and false negatives.

Stage 3 Method

As shown in Figure 20, the dAE + OCSVM + DBSCAN combination demonstrates high precision and recall, effectively identifying true positives with minimal false positives. However, its specificity is diminished, indicating difficulties in accurately identifying normal instances. The F1-score demonstrates robust performance, indicating a well-balanced trade-off between precision and recall. The accuracy is reliable, coupled with a consistently low false negative rate (FNR), highlighting its overall effectiveness.

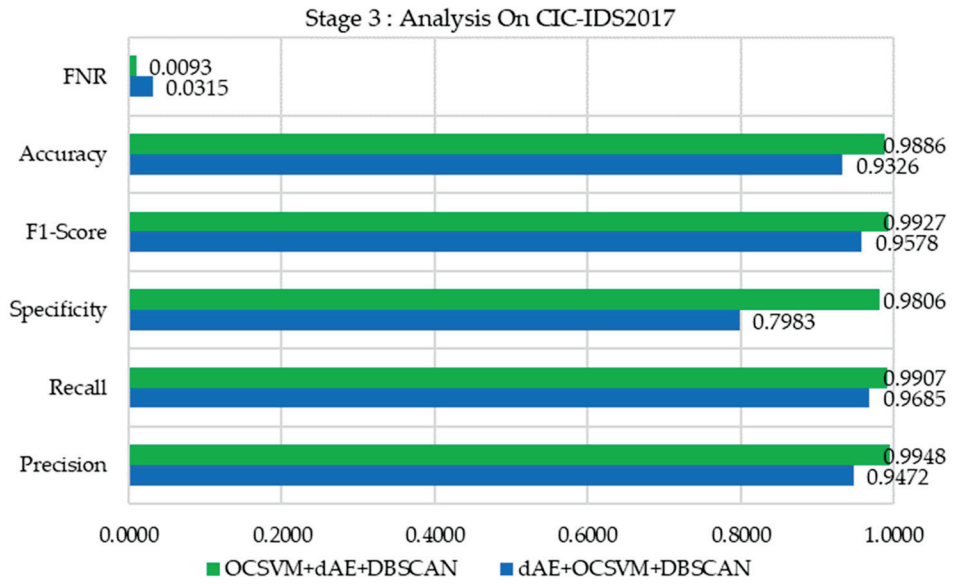


Figure 20. Performance analysis of stage 3 OCSVM + dAE + DBSCAN on CIC-IDS2017.

The proposed approach, combining OCSVM + dAE + DBSCAN, shows superior performance across all metrics. It demonstrates outstanding precision, recall, and specificity, highlighting its robust ability to accurately detect anomalies as well as normal instances. The F1-score and accuracy excel, with the lowest FNR, showcasing its remarkable effectiveness in reducing both false positives and false negatives, establishing it as the top-performing method in this assessment.

Hence, the proposed detection model successfully identified and classified seven distinct cyber-attacks. Subsequently, the prevention model effectively mitigated four of these detected attacks by promptly blocking them and routing the associated data traffic to the security enforcement gateway, due to their high-priority designation within the Cyber Threat Repository. Concurrently, the remaining three attacks, deemed as lower priority, were redirected to the Incident Log Database for further analysis and subsequent mitigation efforts. These findings highlight the effectiveness of the prevention model, as it successfully mitigated all detected attacks. The priority-based blocking strategy is highly effective, successfully blocking all high-priority attacks, as shown in Table 7. Furthermore, Figure 21 demonstrates an impressive success rate in preventing all detected attacks, emphasizing the resilience of the prevention model. These findings confirm the model's ability to uphold security without any compromises.

Table 7. Performance of MITRE ATT&CK-based prevention model.

Metrics	CIC-IDS2017	CSECIC-IDS2018
Prevention rate (PR)	1.000	1.000
Priority based blocking rate (PBBR)	1.000	1.000

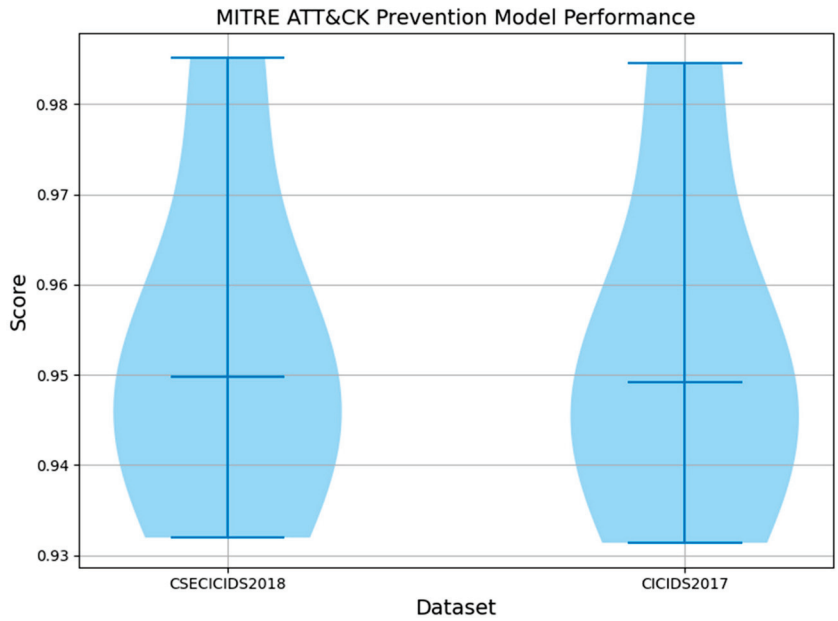


Figure 21. Success rate of MITRE ATT&CK-based prevention model.

The evaluation and results of the proposed cybersecurity solution reveal key insights into its effectiveness and robustness. The assessment of OCSVM underscores the critical influence of the "nu" parameter on detection accuracy, essential for optimizing performance. Similarly, evaluation of the deep autoencoder (dAE) demonstrates its effectiveness in anomaly detection, utilizing MSE across datasets. The detection model's overall performance across both "Normal" and "Attack" data in CIC-IDS2017 and CSECIC-IDS2018 datasets shows significant efficacy, consistently delivering notable results. Moreover, the proposed approach outperforms alternatives in clustering tasks, exhibiting superior precision, recall, and overall accuracy. Comparative analysis across datasets highlights its superiority, with high correctness of clustering, F1-measure, and recall scores. Successful identification and classification of cyber-attacks, coupled with the prevention model's remarkable success rate in mitigating them, underline the solution's robustness in enhancing cybersecurity resilience. Implementation insights elucidate practical deployment strategies, leveraging various techniques like autoencoders, OCSVM, DBSCAN, and the MITRE ATT&CK framework. Overall, these evaluations collectively demonstrate the comprehensive efficacy of the proposed cybersecurity solution in detection, prevention, and overall resilience enhancement, promising in tackling contemporary cybersecurity challenges.

5.7. Limitations and Future Work

Although the proposed method, integrating OCSVM, dAE, and DBSCAN, demonstrates superior performance across most metrics, it does have inherent limitations and shortcomings. The method's efficacy is validated on the CSECIC-IDS2018 and CICIDS2017 datasets. However, its performance may vary on other datasets or in different domains, underscoring the need for additional validation across diverse datasets to ensure broad applicability. Moreover, the effectiveness of the proposed method hinges significantly on meticulously tuning parameters for each algorithm, a task that can consume considerable time. This research focuses on detecting emerging zero-day attacks, utilizing a binary classification algorithm. Future

research can expand on this work by exploring multiclass classification techniques. It should prioritize optimizing the method to minimize computational complexities, validating it across diverse datasets, simplifying parameter tuning, improving scalability, and ensuring reliable performance in varied and dynamic environments.

6. Conclusions

This study presented a novel comprehensive approach to address the escalating cybersecurity concerns associated with the proliferation of Internet services and the corresponding surge in network attacks. By leveraging unsupervised learning techniques, the proposed method offers a robust framework for training detection models to effectively counter a wide range of threats, including denial of service (DoS), distributed denial of service (DDoS), botnet, brute force, infiltration, and Heartbleed. The significance of this work lies in its multi-stage detection model, which combines basic autoencoders (bAEs), a one-class support vector machine (OCSVM), and deep autoencoder (dAE) attack detection, supplemented by density-based spatial clustering of applications with noise (DBSCAN) for attack clustering. This integrated approach enables the accurate delineation of attack clusters, aiding in the mapping of attack tactics and facilitating timely response strategies. Furthermore, by leveraging preprocessed and unlabeled normal network traffic data, this methodology allows for the identification of novel attacks while mitigating the impact of imbalanced training data on model performance. The use of reconstruction error in the autoencoder method, kernel functions in OCSVM, and density-based clustering in DBSCAN ensures robust anomaly detection, noise management, and scalability, resulting in highly accurate intrusion detection. The evaluation of the proposed model on standard datasets such as CIC-IDS2017 and CSECIC-IDS2018 demonstrates its efficacy, with accuracies exceeding 98% in both cases. These results underscore the potential of this approach to significantly enhance network security and protect against evolving cyber threats. This research contributes to the advancement of intrusion detection systems and underscores the importance of leveraging innovative techniques to safeguard critical networks against cyber threats. By combining machine learning algorithms with robust clustering methods, this approach offers a promising solution for efficient and effective intrusion detection in the ever-changing cybersecurity landscape.

Looking ahead, the findings of this study suggest various avenues for application and extension. The developed framework can be further refined and customized to address specific network environments and threat landscapes. Additionally, the integration of frameworks like the MITRE ATT&CK provides a structured approach to cataloging attacks and tactics, enabling proactive defense strategies based on threat prioritization.

Author Contributions: Conceptualization, P.K., S.P., M.T., B.B. and F.B.; methodology, P.K., S.P. and M.T.; software, P.K., S.P. and M.T.; validation, P.K., S.P., M.T., B.B. and F.B.; formal analysis, P.K., S.P. and M.T.; investigation, P.K., S.P., M.T., B.B. and F.B.; resources, P.K., S.P. and M.T.; data curation, P.K., S.P. and M.T.; writing—original draft preparation, P.K., S.P. and M.T.; writing—review and editing, P.K., S.P., M.T., B.B. and F.B.; visualization, B.B. and F.B.; supervision, B.B. and F.B.; project administration, B.B. and F.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset used for this article is available online at https://www.unb.ca/cic/datasets/ids-2018.html and https://www.unb.ca/cic/datasets/ids-2017.html for CSECIC-IDS2018 and CIC-IDS2017 respectively, date accessed 11 July 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Jiang, K.; Wang, W.; Wang, A.; Wu, H. Network Intrusion Detection Combined Hybrid Sampling with Deep Hierarchical Network. *IEEE Access* **2020**, *8*, 32464–32476. [CrossRef]
2. Gandi, V.P.; Jatla, N.S.L.; Sadhineni, G.; Geddamuri, S.; Chaitanya, G.K.; Velmurugan, A.K. A Comparative Study of AI Algorithms for Anomaly-based Intrusion Detection. In Proceedings of the 7th International Conference on Computing Methodologies and Communication, ICCMC 2023, Erode, India, 23–25 February 2023; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2023; pp. 530–534. [CrossRef]
3. Ali, T.A.J.; Jawhar, M.M.T. Detecting network attacks model based on a convolutional neural network. *Int. J. Electr. Comput. Eng.* **2023**, *13*, 3072–3078. [CrossRef]
4. Lv, Z.; Chen, D.; Cao, B.; Song, H.; Lv, H. Secure Deep Learning in Defense in Deep-Learning-as-a-Service Computing Systems in Digital Twins. *IEEE Trans. Comput.* **2024**, *73*, 656–668. [CrossRef]
5. Sun, N.; Ding, M.; Jiang, J.; Xu, W.; Mo, X.; Tai, Y.; Zhang, J. Cyber Threat Intelligence Mining for Proactive Cybersecurity Defense: A Survey and New Perspectives. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 1748–1774. [CrossRef]
6. Pitafi, S.; Anwar, T.; Widia, I.D.M.; Yimwadsana, B.; Pitafi, S. Revolutionizing Perimeter Intrusion Detection: A Machine Learning-Driven Approach with Curated Dataset Generation for Enhanced Security. *IEEE Access* **2023**, *11*, 106954–106966. [CrossRef]
7. Zheng, J.; Zhang, Z.; Ma, Q.; Gao, X.; Tian, C.; Chen, G. Multi-Resource VNF Deployment in a Heterogeneous Cloud. *IEEE Trans. Comput.* **2022**, *71*, 81–91. [CrossRef]
8. Mao, Y.; Shang, X.; Liu, Y.; Yang, Y. Joint Virtual Network Function Placement and Flow Routing in Edge-Cloud Continuum. *IEEE Trans. Comput.* **2024**, *73*, 872–886. [CrossRef]
9. Figueiredo, J.; Serrão, C.; de Almeida, A.M. Deep Learning Model Transposition for Network Intrusion Detection Systems. *Electronics* **2023**, *12*, 293. [CrossRef]
10. Sarhan, M.; Kulatilleke, G.; Lo, W.W.; Layeghy, S.; Portmann, M. DOC-NAD: A Hybrid Deep One-class Classifier for Network Anomaly Detection. *arXiv* **2022**, arXiv:2212.07558.
11. Devarakonda, A.; Sharma, N.; Saha, P.; Ramya, S. Network intrusion detection: A comparative study of four classifiers using the NSL-KDD and KDD'99 datasets. *J. Phys. Conf. Ser.* **2022**, *2161*, 12043. [CrossRef]
12. Wang, C.; Sun, Y.; Lv, S.; Wang, C.; Liu, H.; Wang, B. Intrusion Detection System Based on One-Class Support Vector Machine and Gaussian Mixture Model. *Electronics* **2023**, *12*, 930. [CrossRef]
13. Ren, Y.; Feng, K.; Hu, F.; Chen, L.; Chen, Y. A Lightweight Unsupervised Intrusion Detection Model Based on Variational Auto-Encoder. *Sensors* **2023**, *23*, 8407. [CrossRef] [PubMed]
14. Jain, P.; Bajpai, M.S.; Pamula, R. A Modified DBSCAN Algorithm for Anomaly Detection in Time-series Data with Seasonality. *Int. Arab. J. Inf. Technol.* **2022**, *19*, 23–28. [CrossRef] [PubMed]
15. Xiong, W.; Legrand, E.; Åberg, O.; Lagerström, R. Cyber security threat modeling based on the MITRE Enterprise ATT&CK Matrix. *Softw. Syst. Model.* **2022**, *21*, 157–177. [CrossRef]
16. Sokkalingam, S.; Ramakrishnan, R. An intelligent intrusion detection system for distributed denial of service attacks: A support vector machine with hybrid optimization algorithm based approach. *Concurr. Comput.* **2022**, *34*, e7334. [CrossRef]
17. Duhayyim, M.A.; Alissa, K.A.; Alrayes, F.S.; Alotaibi, S.S.; Tag El Din, E.M.; Abdelmageed, A.A.; Yaseen, I.; Motwakel, A. Evolutionary-Based Deep Stacked Autoencoder for Intrusion Detection in a Cloud-Based Cyber-Physical System. *Appl. Sci.* **2022**, *12*, 6875. [CrossRef]
18. Mousa, A.K.; Abdullah, M.N. An Improved Deep Learning Model for DDoS Detection Based on Hybrid Stacked Autoencoder and Checkpoint Network. *Future Internet* **2023**, *15*, 278. [CrossRef]
19. Shin, C.; Lee, I.; Choi, C. Exploiting TTP Co-Occurrence via GloVe-Based Embedding with MITRE ATT&CK Framework. *IEEE Access* **2023**, *11*, 100823–100831. [CrossRef]
20. Liu, H.; Lang, B. Machine learning and deep learning methods for intrusion detection systems: A survey. *Appl. Sci.* **2019**, *9*, 4396. [CrossRef]
21. Thirimanne, S.P.; Jayawardana, L.; Yasakethu, L.; Liyanaarachchi, P.; Hewage, C. Deep Neural Network Based Real-Time Intrusion Detection System. *SN Comput. Sci.* **2022**, *3*, 145. [CrossRef]
22. Guarino, S.; Vitale, F.; Flammini, F.; Faramondi, L.; Mazzocca, N.; Setola, R. A Two-Level Fusion Framework for Cyber-Physical Anomaly Detection. *IEEE Trans. Ind. Cyber-Phys. Syst.* **2024**, *2*, 1–13. [CrossRef]
23. Ramasamy, M.; Eric, P.V. A novel classification and clustering algorithms for intrusion detection system on convolutional neural network. *Bull. Electr. Eng. Inform.* **2022**, *11*, 2845–2855. [CrossRef]
24. Ahmad, Z.; Khan, A.S.; Shiang, C.W.; Abdullah, J.; Ahmad, F. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Trans. Emerg. Telecommun. Technol.* **2021**, *32*, e4150. [CrossRef]
25. Fan, J.F.J.; Fan, G.Y.J.; Yang, J.G.G. DDoS Attack Detection System Based on RF-SVM-IL Model Under SDN. *J. Comput. Sci.* **2021**, *32*, 031–043. [CrossRef]
26. Yaras, S.; Dener, M. IoT-Based Intrusion Detection System Using New Hybrid Deep Learning Algorithm. *Electronics* **2024**, *13*, 1053. [CrossRef]
27. Harahsheh, K.; Al-Naimat, R.; Chen, C.H. Using Feature Selection Enhancement to Evaluate Attack Detection in the Internet of Things Environment. *Electronics* **2024**, *13*, 1678. [CrossRef]

28. Javed, A.; Ehtsham, A.; Jawad, M.; Awais, M.N.; Qureshi, A.-H.; Larijani, H. Implementation of Lightweight Machine Learning-Based Intrusion Detection System on IoT Devices of Smart Homes. *Future Internet* **2024**, *16*, 200. [CrossRef]
29. Liao, J.; Teo, S.G.; Kundu, P.P.; Truong-Huu, T. ENAD: An ensemble framework for unsupervised network anomaly detection. In Proceedings of the 2021 IEEE International Conference on Cyber Security and Resilience, CSR 2021, Rhodes, Greece, 26–28 July 2021; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2021; pp. 81–88. [CrossRef]
30. Almaraz-Rivera, J.G.; Cantoral-Ceballos, J.A.; Botero, J.F. Enhancing IoT Network Security: Unveiling the Power of Self-Supervised Learning against DDoS Attacks. *Sensors* **2023**, *23*, 8701. [CrossRef]
31. Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J.; Alazab, A. Hybrid intrusion detection system based on the stacking ensemble of C5 decision tree classifier and one class support vector machine. *Electronics* **2020**, *9*, 173. [CrossRef]
32. Shafin, S.S.; Karmakar, G.; Mareels, I. Obfuscated Memory Malware Detection in Resource-Constrained IoT Devices for Smart City Applications. *Sensors* **2023**, *23*, 5348. [CrossRef]
33. Ravi, N.; Shalinie, S.M. Semisupervised-Learning-Based Security to Detect and Mitigate Intrusions in IoT Network. *IEEE Internet Things J.* **2020**, *7*, 11041–11052. [CrossRef]
34. Li, W.; Meng, W.; Au, M.H. Enhancing collaborative intrusion detection via disagreement-based semi-supervised learning in IoT environments. *J. Netw. Comput. Appl.* **2020**, *161*, 102631. [CrossRef]
35. Kwon, R.; Ashley, T.D.; Castleberry, J.E.; McKenzie, P.L.; Gourisetti, S.N.G. Cyber Threat Dictionary Using MITRE ATT&CK Matrix and NIST Cybersecurity Framework Mapping. United States 2020. Available online: https://www.osti.gov/biblio/1734565 (accessed on 17 July 2024).
36. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In Proceedings of the ICISSP 2018—The 4th International Conference on Information Systems Security and Privacy, Madeira, Portugal, 22–24 January 2018; SciTePress: Setúbal, Portugal, 2018; pp. 108–116. [CrossRef]
37. Mustafa, D.H.; Husien, I.M. Adaptive DBSCAN with Grey Wolf Optimizer for Botnet Detection. *Int. J. Intell. Eng. Syst.* **2023**, *16*, 409–421. [CrossRef]
38. Nguyen, T.L.; Kao, H.; Nguyen, T.T.; Horng, M.F.; Shieh, C.S. Unknown DDoS Attack Detection with Fuzzy C-Means Clustering and Spatial Location Constraint Prototype Loss. *Comput. Mater. Contin.* **2024**, *78*, 2181–2205. [CrossRef]
39. Dwivedi, D.; Bhushan, A.; Singh, A.K.; Snehlata. Leveraging K-means clustering for enhanced detection of network traffic attacks. In Proceedings of the 2024 3rd International conference on Power Electronics and IoT Applications in Renewable Energy and its Control (PARC), Mathura, India, 23–24 February 2024; pp. 72–76. [CrossRef]
40. An, H.; Ma, R.; Yan, Y.; Chen, T.; Zhao, Y.; Li, P.; Li, J.; Wang, X.; Fan, D.; Lv, C. Finsformer: A Novel Approach to Detecting Financial Attacks Using Transformer and Cluster-Attention. *Appl. Sci.* **2024**, *14*, 460. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Article

Exploiting Autoencoder-Based Anomaly Detection to Enhance Cybersecurity in Power Grids

Fouzi Harrou [1,*], Benamar Bouyeddou [2], Abdelkader Dairi [3,*] and Ying Sun [1]

[1] Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia; ying.sun@kaust.edu.sa
[2] LESM Laboratory, Department of Telecommunications, Faculty of Technology, University of Saida-Dr Moulay Tahar, Saida 20000, Algeria; benamar.bouyeddou@univ-saida.dz
[3] Computer Science Department, University of Science and Technology of Oran-Mohamed Boudiaf (USTO-MB), El Mnaouar, BP 1505, Oran 31000, Algeria
* Correspondence: fouzi.harrou@kaust.edu.sa (F.H.); abdelkader.dairi@univ-usto.dz (A.D.)

Abstract: The evolution of smart grids has led to technological advances and a demand for more efficient and sustainable energy systems. However, the deployment of communication systems in smart grids has increased the threat of cyberattacks, which can result in power outages and disruptions. This paper presents a semi-supervised hybrid deep learning model that combines a Gated Recurrent Unit (GRU)-based Stacked Autoencoder (AE-GRU) with anomaly detection algorithms, including Isolation Forest, Local Outlier Factor, One-Class SVM, and Elliptical Envelope. Using GRU units in both the encoder and decoder sides of the stacked autoencoder enables the effective capture of temporal patterns and dependencies, facilitating dimensionality reduction, feature extraction, and accurate reconstruction for enhanced anomaly detection in smart grids. The proposed approach utilizes unlabeled data to monitor network traffic and identify suspicious data flow. Specifically, the AE-GRU is performed for data reduction and extracting relevant features, and then the anomaly algorithms are applied to reveal potential cyberattacks. The proposed framework is evaluated using the widely adopted IEC 60870-5-104 traffic dataset. The experimental results demonstrate that the proposed approach outperforms standalone algorithms, with the AE-GRU-based LOF method achieving the highest detection rate. Thus, the proposed approach can potentially enhance the cybersecurity in smart grids by accurately detecting and preventing cyberattacks.

Keywords: cyberattack detection; protocol IEC 104; deep learning; semi-supervised methods; anomaly detection

1. Introduction

With the rapid digitization of energy infrastructure, cybersecurity has become a critical concern for smart grids [1]. Due to their interconnected nature, the use of advanced communication and information technologies to enhance efficiency, reliability, and sustainability renders smart grids increasingly vulnerable to cyber threats. Smart grids, envisioned as the future of power grids, integrate advanced technologies such as sensors, automation, and communication systems for real-time monitoring and control [2,3]. They facilitate quicker problem identification and support demand response programs, renewable energy integration, and operation in diverse environments with adaptable technologies like Industrial Internet of Things (IIoT), cloud computing, and Software-Defined Networking (SDN). Components such as sensors, meters, controllers, and communication networks provide numerous entry points for potential attackers, leading to severe consequences such as power disruptions and the compromise of critical infrastructure. Ensuring the integrity and resilience of smart grid systems necessitates innovative cybersecurity measures tailored to the evolving threat landscape [4].

Nevertheless, the growing dependence on vulnerable advanced communication systems and highly connected devices has heightened smart grids' vulnerability to cyberattacks and malicious activities [5]. As a result, cyber attackers have a broadened attack surface within the smart grid ecosystem, targeting insecure communication networks, outdated software, weak authentication, and other vulnerabilities [3,5]. Indeed, a successful cyberattack on a smart grid can cause major disruptions to power generation and distribution, endangering public safety via widespread outages. Moreover, such attacks can compromise sensitive information's confidentiality, integrity, and availability, including customer data, financial records, and critical infrastructure data [6]. Various types of cyberattacks can be launched against smart grids, including Man-in-the-middle (MitM), False data injection, DoS/DDoS (Denial of Service/Distributed Denial of Service), Insider attacks, Phishing attacks, Malware attacks, and physically damaging or sabotaging smart grid components. For example, significant cyberattacks occurred in recent years, targeting Iran's nuclear centrifuges (2010), the Ukrainian power system (2015), the US power grid (2019), Energie de Portugal (2020), and a Ukrainian energy company (2022) [5,7].

Ensuring cybersecurity is essential for the successful deployment and operation of smart grids. This includes implementing security protocols, deploying intrusion detection and prevention systems, training personnel on cybersecurity best practices, and continuously monitoring and updating the smart grid systems. Consequently, machine learning (ML) has emerged as a popular method to detect and prevent cyberattacks in smart grids [8]. ML algorithms analyze vast amounts of operational data and identify unusual behaviors in network traffic, enabling them to detect potential security threats and alert the system administrators. As a result, ML plays a crucial role in maintaining the security and reliability of smart grid systems. Various ML-based approaches, including supervised, unsupervised, and semi-supervised learning, have been researched to defend against different types of cyberattacks. For example, An et al. [9] devised a detection approach using a deep-Q-network (DQN) to safeguard AC power systems against data integrity attacks (DIA). Their study showed that the DQN-based detection approach surpassed the standard methods in regard to accuracy and speed. Sakhnini et al. [10] examined three distinct algorithms—Artificial Neural Networks (ANNs), k-Nearest Neighbor (KNN), and Support Vector Machine (SVM)—incorporating various feature selection (FS) techniques, particularly the genetic algorithm (GA), which proved to be the most efficient FS method. Hink et al. [11] explored ML-based techniques to differentiate among regular operations, cyber-attacks, and natural disturbances in smart grids, evaluating algorithms like Random Forest (RF), Naïve Bayes (NB), SVM, and AdaBoost. Esmalifalak et al. [12] tackled stealthy false-data injection attacks using Principal Component Analysis (PCA)-based dimensionality reduction and SVM trained over labeled data for detection. Ahmed et al. [13] introduced an unsupervised Isolation Forest (iForest) algorithm for detecting covert data-integrity exploits using unlabeled historical data of State Estimation Measurement Features (SE-MF) and a PCA-based data reduction technique. Acosta et al. [14] addressed the detection of stealthy attacks in large-scale smart grid networks by applying Kernel PCA (KPCA) to solve the computational complexity issues. The output data from KPCA were fed into the Extra-Trees algorithm. Al-Qudah et al. [15] presented an integrated solution combining one-class classification (OCSVM) with PCA for the accurate detection of memory dump malware, including novel types. Menon et al. [16] proposed an unsupervised K-means-based model to identify Denial of Service (DoS) attacks, such as UDP and ICMP flood attacks, directed at the AMI network. The model, relying on unlabeled data, successfully clustered both normal and abnormal behaviors.

The evolution of smart grids has spurred technological advancements and an increasing demand for more efficient and sustainable energy systems. However, the integration of communication systems into smart grids has escalated the risk of cyberattacks, potentially leading to power outages and system disruptions. This paper introduces a semi-supervised approach for detecting malicious attacks in Industrial Control Systems (ICSs) and Supervisory Control and Data Acquisition (SCADA) networks. Unlike traditional supervised

learning, which relies on labeled data, this method utilizes only attack-free data for training. This eliminates the need for costly and time-consuming manual labeling, making it more practical and cost-effective for real-world deployment. By learning from attack-free data, the model can effectively identify deviations from normal behavior during testing, without the need for explicit labels. This approach simplifies the training process and reduces the risk of bias introduced by human labeling errors. The contributions of this study are summarized next.

- Firstly, a hybrid deep learning approach is proposed for detecting the cyberattacks in smart grids. Specifically, we introduce the Autoencoder (AE)-Gated Recurrent Unit (GRU) approach, integrating two stacked GRU layers into the AE model's encoder and decoder. By incorporating GRU units into both the encoder and decoder sides, the model learns compact representations of sequential smart grid data while ensuring minimal loss in data reconstruction. For cyberattack detection, the AE-GRU approach is combined with semi-supervised anomaly detection methods, including Isolation Forest (IF), Local Outlier Factor (LOF), OCSVM, and Elliptical Envelope (EE). Initially, the AE-GRU approach extracts pertinent features from the data, seamlessly integrated with semi-supervised techniques (IF, OCSVM, EE, and LOF) to detect anomalies effectively. Specifically, AE-GRU handles data reduction and feature extraction, followed by applying anomaly detection algorithms to identify potential cyberattacks.
- The effectiveness of these approaches was evaluated based on the IEC 60870-5-104 (IEC 104) control communication protocol, commonly used in SCADA networks to manage critical infrastructures such as power stations. The IEC 104 protocol is widely used in smart grid environments for the exchange of control and monitoring information between various devices and systems. This protocol facilitates the communication between different components of the power grid, including sensors, actuators, controllers, and SCADA systems. The proposed framework's effectiveness was assessed using the widely adopted IEC 60870-5-104 traffic dataset, which includes various types of malicious attacks such as DOS, Switch, Injection, ConnLoss, and Rogue Dev. We employed four statistical indices—accuracy, precision, F1-score, and the Area Under the Curve (AUC)—to compare the discrimination accuracy of the considered methods. The experimental results demonstrate that the proposed approach outperforms standalone algorithms, with the AE-GRU-based LOF method achieving the highest detection rate. This underscores the potential for the proposed approach to enhance the cybersecurity in smart grids by accurately detecting and mitigating cyber threats.

The remaining sections of this paper are structured as follows: Section 2 discusses the related works on deep learning-based detection approaches for cyberattacks in smart grids. Section 3 provides a detailed description of the proposed framework. The performance evaluation experiments and detection results are presented in Section 4. The paper concludes in Section 5, followed by an outline of future work.

2. Related Works

DL algorithms have proven effective across various domains, including image recognition, speech processing, and anomaly detection. DL effectively analyzes data from sensors, control systems, and communication networks within smart grids. Recent interest focuses on using DL for cyberattack detection in smart grids, with studies exploring its application [17–19]. This section provides an overview of DL-based attack detection in smart grids.

In [20], Haghshenas et al. introduced a Temporal Graph Neural Network (TGNN) architecture to detect FDI and ramp attacks. It constructs a temporal graph, utilizes a three-layer GNN with GRU units to learn normal behavior, and employs a graph attention mechanism for attack localization. However, the model's assumption of temporal sequencing in cyberattacks may not always hold, posing challenges for detection and localization. Khanna et al. [21] introduced an approach to detect compromised meters in smart grids using the KLD (Kullback–Leibler distance), ANNs (Artificial Neural Networks), and ELMs

(Extreme Learning Machines). The KLD quantifies the dissimilarity between historical and real-time measurements to identify anomalies that signal data integrity attacks. ANNs and ELMs accurately predict the current loads and identify compromised meters based on an error threshold. In [22], Karimipour et al. introduced an unsupervised model-free method based on Symbolic Dynamic Filtering (SDF) for feature extraction, a Dynamic Bayesian Network for transition probabilities, RBM for learning behavior patterns, and a Relative Entropy (RE) metric for anomaly detection. The accuracy of the framework may vary due to different symbolization methods for time series data. Qiu et al. [23] introduced the SG-FST-MCNN, a multi-view convolutional neural network (MCNN), for authenticating DSD (Distribution Synchrophasor Data) sources and detecting spoofing attacks. It utilizes Savitzky–Golay (SG) filter and fast S transform (FST) for feature extraction, followed by MCNN for classification without manual feature selection. However, an SG filter and FST assume linearity, which may limit their effectiveness with non-linear data. The study in [24] introduced a two-stage deep learning framework utilizing image recognition models to detect and locate attacks in multivariate time-series data. The framework converts data into 2D images using the Gramian Angular Field (GAF) and Recurrence Plot (RP) techniques and trains a multi-label CNN classifier for attack detection and localization. However, this approach is computationally demanding, presenting challenges for real-time detection.

Shan et al. [25] targeted DDOS attacks using the Autoencoder (AE) algorithm. They utilized a multi-level mixture of shallow and deep marginalized stacked denoising autoencoders (MSDA) for feature extraction. These features were projected into a kernel space, where the Multiple Kernel Learning for Dimensionality Reduction (MKLDR) algorithm trained the detection model. However, MKLDR faces challenges in selecting suitable kernels and has limited scalability for large or high-dimensional datasets. Dou et al. [26] proposed a detection method that combines Variational Mode Decomposition (VMD) and OS-Extreme Learning Machines (OSELMs) to identify abnormal power system states. This approach extracts multiscale modes and analyzes statistical features to assess irregular behaviors. An OSELM correlates these features with the power system security status and identifies abnormal states. However, the effectiveness of the F-test may be limited by assumptions, particularly the handling of outliers and non-normal data distributions in real-world settings. Lu et al. [27] focused on intrusion detection in Advanced Metering Infrastructure (AMI) using a Stacked Autoencoder (SAE) and Long Short-Term Memory (LSTM) models. A SAE reduced the input feature dimensionality, while the Bi-directional LSTM (BiLSTM) identified abnormal traffic patterns. An attention mechanism assigns weight coefficients to important features, enhancing the classification accuracy. Qi et al. [28] exploited Phasor Measurement Unit (PMU) data to detect cyberattacks by employing the deep autoencoder (DAE) and semi-supervised anomaly detection techniques. Such an approach may not be effective when changing the data distribution or when there is insufficient information in unlabeled data. Ferrag et al. [29] developed DeepCoin, integrating blockchain with a recurrent neural network (RNN) to secure smart grid transactions while preserving privacy. DeepCoin employs practical Byzantine fault tolerance for blockchain consensus and RNN to verify security requirements and prevent cyberattacks.

3. AE-GRU Anomaly Detection Framework

In this section, an overview of the materials needed to develop the proposed AE-GRU-based detection scheme is provided, including the Autoencoder, GRU, and semi-supervised anomaly detection methods.

3.1. Auto-Encoders

An autoencoder, a type of neural network architecture widely employed in tasks such as dimensionality reduction, feature learning, and data compression, is particularly valuable in cybersecurity applications [30]. An autoencoder comprises two main components: an encoder network and a decoder network (Figure 1). The encoder network compresses

input data into a latent space representation, whereas the decoder network reconstructs the original input based on this compressed representation.

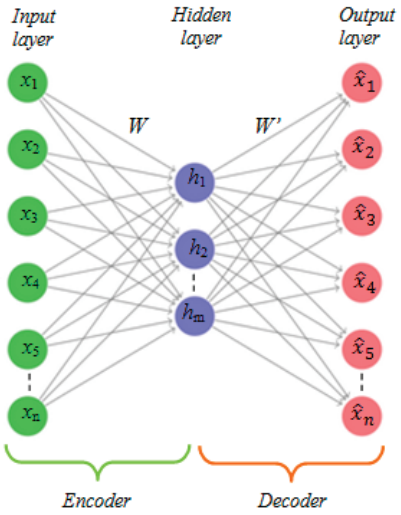


Figure 1. Fundamental structure of an autoencoder (AE).

The encoding stage of an autoencoder is pivotal in compressing input data, $x = \{x_1, x_2, \ldots, x_n\}$, into a lower-dimensional latent space representation, $h = \{h_1, h_2, \ldots, h_m\}$, which contains essential features and patterns for efficient reconstruction by the decoder stage. This transformation involves applying a linear transformation to the input, followed by a non-linear activation function, represented mathematically as [31]:

$$h = f(x) = A_f(Wx + b), \tag{1}$$

where W denotes the weight matrix, b represents the bias vector, and A_f is the activation function applied element-wise. In the decoding stage, the compressed representation h from the encoding stage is transformed back into the original input data space. This process involves applying a series of hidden layers to h, each involving a linear transformation followed by a non-linear activation function, represented mathematically as [31,32]:

$$\hat{x} = g(h) = A_g(W'h + b'), \tag{2}$$

where W' is the weight matrix, b' is the bias vector, and A_g is the activation function. The output of this transformation, $\hat{x} = \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n\}$, represents the reconstructed input data. The goal of the decoding stage is to generate a reconstruction that closely matches the original input data x, thereby capturing the essential features and patterns encoded in the compressed representation h.

During training, an AE adjusts the parameters such as weights and biases to minimize the difference between the input data and their reconstructed output. This iterative process utilizes optimization algorithms like the stochastic gradient descent (SGD) algorithm to update the parameters and minimize a chosen loss function. Cross-entropy is widely used as a loss function in autoencoders for its effectiveness in capturing the differences between the input and reconstructed data distributions. It is expressed as [32]:

$$L_{BCE} = \frac{1}{M}\sum_{i=1}^{m}\sum_{j=1}^{n}\left[x_{j,i} log \hat{x}_{j,i} + (1 - x_{j,i}) log(1 - \hat{x}_{j,i})\right], \tag{3}$$

where M represents the number of training samples or instances, and $x_{j,i}$ denotes the j-th feature of the i-th training instance. In cybersecurity, autoencoders trained on normal data learn patterns and structures. Deviations from these learned patterns during inference can signal anomalies or cyber threats.

3.2. GRUs (Gated Reccurent Units)

A GRU is a type of recurrent neural network (RNN) designed to mitigate the vanishing gradient problem commonly encountered in traditional RNNs [33]. Introduced by Cho et al. in 2014, GRUs employ gating mechanisms to selectively control information flow, reducing the risk of gradient vanishing or exploding during training [34]. Specifically, GRUs consist of two main components: the update gate and the reset gate (Figure 2). The update gate determines how much of the previous hidden state to retain and how much of the new input to incorporate, while the reset gate determines how much of the previous hidden state to forget and how much of the new input to focus on [35].

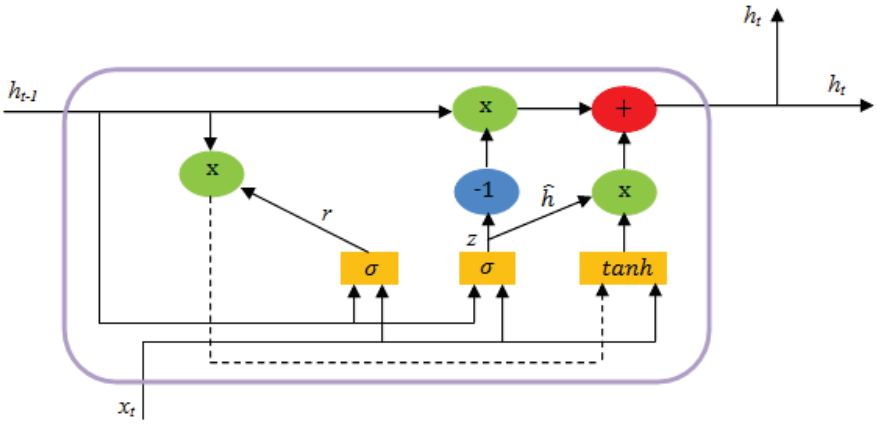


Figure 2. General representation of a GRU.

For the input data $x_t = \{x_1, x_2, \ldots, x_n\}$ and state $h_t = \{h_1, h_2, \ldots, h_m\}$ vectors at time t, the update gate z_t is computed as [36]:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z), \tag{4}$$

where W_z and U_z represent the weight matrix, and σ is the activation sigmoid function. The reset gate r_t in Equation (8) controls how much of the previous hidden state to forget [36].

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r), \tag{5}$$

These gates are combined using element-wise multiplication to determine the updated hidden state. The output of the GRU is then computed based on a combination of the current input and the updated hidden state [36]:

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \hat{h}_t, \tag{6}$$

$$\hat{h}_t = \tanh(W_h x_t + U_h(h_{t-1} \odot r_t)), \tag{7}$$

where h_t is the hidden state at time step *t* and $\hat{h}$ is the candidate hidden state, representing the proposed update to the current hidden state. GRUs typically provide similar capabilities to LSTM units but with fewer parameters, ensuring computational efficiency while effectively capturing the long-term dependencies in sequential data.

3.3. Hybrid AE-GRU

In this section, we introduce the hybrid AE-GRU model, which combines the advantages of AE and GRU architectures to improve the cybersecurity measures in smart grids. The AE-GRU model is designed to encode time-series data effectively, aiming to mitigate the potential cyber threats. The AE-GRU model integrates two stacked GRUs within both the encoder and decoder components (Figure 3). This configuration enables the model to capture the intricate temporal patterns and anomalies present in smart grid data. By leveraging the sequential nature of GRUs, the model can efficiently extract features from the

input data and reconstruct normal traffic patterns. The primary objective of the AE-GRU model is to minimize the reconstruction error, typically measured using metrics like Binary Cross-Entropy (BCE) [31]. This metric evaluates the dissimilarity between the original input data and the reconstructed output. By reducing the reconstruction error, the AE-GRU model enhances its ability to accurately capture and represent the underlying structure of the smart grid data.

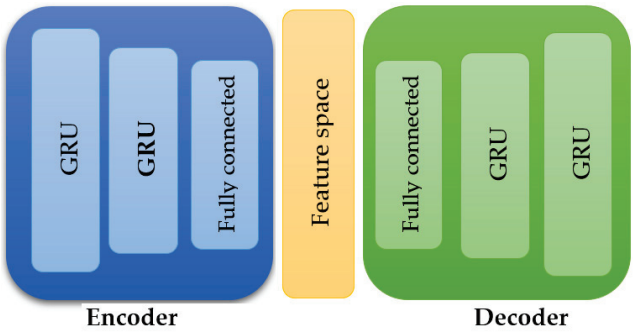


Figure 3. Architecture of the proposed hybrid AE-GRU model.

The integration of the AE-GRU model represents a proactive measure to enhance the cybersecurity in smart grids. By effectively encoding the time-series data, the model can be used in conjunction with semi-supervised anomaly detection methods to identify potential cyber threats, such as malicious attacks or anomalies in system behavior.

3.4. Semi-Supervised Algorithms

The primary objective of the detection stage is to differentiate between regular and unusual traffic routed in the smart grid. In this study, four semi-supervised anomaly detection techniques were explored: a One-Class Support Vector Machine (1SVM), Local Outlier Factor (LOF), Isolation Forest (iF), and Elliptic Envelope (EE). These methods are designed to detect the anomalies in data without requiring labeled instances of anomalies during training. In other words, these methods were developed using a semi-supervised training approach, which involves training only with normal data (i.e., attack-free data).

- A One-Class SVM is a machine learning algorithm used for anomaly detection, particularly in cases where only normal data are available for training. It works by constructing a hyperplane in a high-dimensional space that encapsulates the normal data points. This hyperplane should ideally separate the normal data from the rest of the feature space which contains anomalies [37,38]. During training, the algorithm learns to maximize the margin around the normal data points while minimizing the possibility of including anomalies within this margin. During inference, data points lying outside this margin are classified as anomalies [39].
- Isolation Forest (iF) is an ensemble learning method designed for outlier detection. It constructs a set of isolation trees, each of which partitions the feature space by randomly selecting a feature and a split value. Anomalies are expected to be isolated in fewer partitions compared to the normal data points [40]. Therefore, they require fewer splits to be separated from the majority of the data. The algorithm measures the average path length for each data point in the trees. The shorter average path lengths indicate anomalies, while the longer path lengths suggest normal data [41].
- An Elliptic Envelope (EE) is a statistical method used for multivariate outlier detection assuming that the normal data points are located within a tightly bound elliptical region. The algorithm fits an ellipse to the data using robust covariance estimation techniques [42]. Data points lying outside the ellipse are considered outliers, as they deviate significantly from the expected distribution of normal data. The EE method is particularly useful when the normal data follow a Gaussian distribution and when the dimensionality of the data is relatively low [41].

- The LOF method comprises a density-based anomaly detection algorithm that measures the local density deviation of a data point with respect to its neighbors. It operates under the assumption that anomalies have lower densities compared to their neighbors, resulting in higher LOF scores [43]. The LOF score of a data point is computed based on the ratio of the local density of the point to the average local density of its neighbors. Data points with significantly higher LOF scores compared to their neighbors are considered outliers.

Table 1 provides a concise overview of the characteristics of these four anomaly detection methods. These approaches offer different strategies for identifying the anomalies in data and can be applied depending on the specific characteristics of the dataset and the nature of the anomalies being targeted.

Table 1. Comparison of the four semi-supervised anomaly detection methods.

Approach	Basic Idea	Pros	Cons
1SVM	Constructs a hyperplane to encapsulate normal data and separates it from anomalies.	- Effective for high-dimensional data - Works well with small training data	- Sensitive to parameter tuning - Slow training on large datasets
LOF	Measures the local density deviation of a data point with respect to its neighbors.	- Robust to outliers and noise - Does not assume a specific data distribution	- The computational complexity increases with dataset size - Parameter selection can be challenging
Isolation Forest (iF)	Constructs isolation trees to isolate anomalies with fewer splits than normal data.	- Efficient for high-dimensional data - Fast training and inference	- Prone to overfitting with noisy data - Limited interpretability
Elliptic Envelope (EE)	Fits an ellipse to the data and detects outliers outside the ellipse.	- Robust to multivariate outliers - Effective for low-dimensional data	- Assumes a Gaussian distribution of normal data - Limited applicability to non-Gaussian data

3.5. The Proposed AE-GRU-Driven Anomaly Detection Framework

This section provides a brief overview of the proposed anomaly detection framework for detecting the cyberattacks in smart grids, specifically focusing on monitoring network traffic data, mainly focusing on the IEC 104 protocol communication. This protocol is commonly used for exchanging control and monitoring information between the devices in smart grids. By focusing on IEC 104 data, the framework specifically targets the communication within the smart grid control system, potentially leading to more accurate anomaly detection compared to analyzing general network traffic. To effectively model the distribution of normal traffic, it is imperative to capture the temporal data dependencies

inherent in data sequences and time-series data. The proposed deep learning model is designed to achieve this by integrating the AE with the GRU model, which is renowned for its ability to learn long-range dependencies, into its structure. Within this framework, the autoencoder (AE) plays a pivotal role, performing two critical tasks: feature extraction and dimensionality reduction. Through the training process, the deep AE constructs a compact representation of the data, incorporating the relevant features and reducing the dimensionality. By combining the robustness of deep autoencoders with the effectiveness of recurrent neural networks, the AE-GRU model can effectively capture lengthy time-period dependencies. Importantly, this method is agnostic to the communication standard employed in the smart grid network, allowing for versatility across various standards. For the purposes of this study, the proposed cyberattack detection procedure was evaluated using the IEC-104 standard. The process flow is depicted in Figure 4, illustrating the two phases: training and detection. Data preprocessing involves the removal of missing values and standardization of numeric features. Subsequently, during the training phase, the AE-GRU model and anomaly detection procedures are trained using attack-free data. The trained model is then evaluated using testing data to assess its performance.

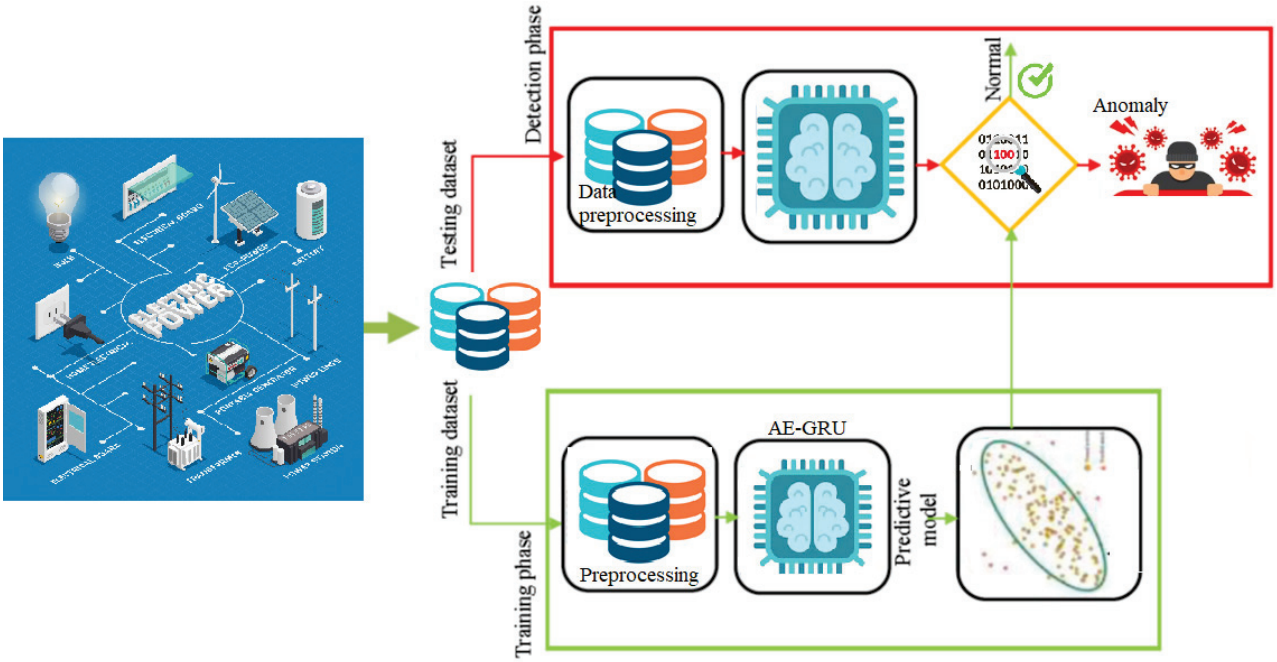


Figure 4. Flowchart of the AE-GRU based detection framework.

During the training stage, the data are normalized and organized into sequences of specific lengths. Encoding these sequences generates a compact representation containing the essential features, with the training process aiming to minimize the reconstruction error using the cross-entropy loss function. This iterative process continues until the model converges and the reconstruction error stabilizes. The resulting feature space contains normal traffic communications without any attacks, which is utilized to train the anomaly detection methods considered in this study.

The anomaly detection methods are trained on the feature space obtained from encoding the normal data. When a data sequence corresponds to a specific attack category, its numerical signature is encoded using the model parameters learned from the normal data. Consequently, any deviation from the encoding of the normal data can be detected by the anomaly detectors.

The performance of the designed framework was assessed using four widely used statistical metrics: Accuracy, Precision, Recall, and F1-score. In a binary detection scenario,

these metrics are calculated based on the number of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \tag{8}$$

$$Recall = \frac{TP}{TP + FN}, \tag{9}$$

$$Precision = \frac{TP}{TP + FP}, \tag{10}$$

$$F1 - score = 2\frac{Precision - Recall}{Precison + Recall} = \frac{2TP}{2TP + FP + FN}, \tag{11}$$

Before presenting the results, the next section provides an overview of the IEC 608705-104 protocol and its significance in SCADA systems, particularly in smart grid networks.

3.6. An Overview of the IEC-60780-5-104 Protocol

The IEC 60870-5-104 (IEC 104) protocol is a fundamental component of SCADA networks, which is pivotal in managing and overseeing critical infrastructures such as power stations [44]. Designed to enable seamless communication between control centers and substations, IEC 104 facilitates real-time monitoring, control, and data acquisition within power grid systems. Its adoption has become widespread due to its ability to ensure the reliable and efficient transmission of data, essential for maintaining the stability and functionality of power distribution networks. Unlike its predecessor, the IEC 60870-5-101 protocol, IEC 104 operates at higher speeds, allowing for faster and more responsive communication between substations and control centers. This enhanced speed is crucial for meeting the demands of the modern power grid operations, where rapid decision-making and real-time data exchange are paramount.

The architecture of the IEC 104 protocol follows a client–server model, where substations act as servers and control centers act as clients. This structure enables bidirectional communication, facilitating the exchange of control commands, status information, and measurement data between the different components of the power grid infrastructure [45]. IEC 104 relies on standard networking protocols such as the Transmission Control Protocol (TCP) and Internet Protocol (IP) for data transmission. This reliance on widely accepted networking standards ensures the compatibility with the existing network infrastructure and simplifies the integration into modern communication networks. In addition to defining the communication procedures and data formats, the IEC 104 protocol also specifies various data types tailored to the specific requirements of power grid applications. These data types encompass a wide range of parameters, including analog values, binary states, and control commands, allowing for the comprehensive monitoring and control of the power grid operations.

While the IEC 104 protocol offers robust communication capabilities, it is not immune to cybersecurity threats [46]. As the power grid systems become increasingly interconnected and digitized, the protocol's vulnerabilities may be exploited by malicious actors seeking to disrupt operations or compromise the sensitive infrastructure. Therefore, understanding the intricacies of the IEC 104 protocol is essential for developing effective cybersecurity strategies and implementing robust anomaly detection techniques to safeguard the critical power grid assets against potential cyberattacks.

4. Threat Scenarios and Results

This section provides an overview of the utilized 4.1 BUT-IEC104-I Dataset, followed by presenting the detection results based on the AE-GRU-based methods and the considered standalone anomaly detection approaches.

4.1. BUT-IEC104-I Dataset

The BUT-IEC104-I dataset provides a valuable resource for cybersecurity research, particularly for assessing the anomaly detection methods within the context of the IEC 60870-5-104 (IEC 104) communication protocols. Developed by Matousek et al. [47,48] from Brno University of Technology, this dataset consists of network traffic traces collected from a real-world smart grid environment that conforms to the industrial network standard, IEC 608705-104. It simulates the realistic scenarios encountered in power grid operations. The traffic was monitored using an IPFIX sniffer, capturing IPFIX flow and application protocol headers. The dataset includes various traffic characteristics, such as IP addresses, ports, object IDs, as well as attributes like start and end times and data exchange quantity. Each line in the dataset captures a range of headers extracted from the IEC 104 packet, totaling 16 features. These include the timestamp (absolute time) and relative time (in seconds from the start of capturing), as well as source and destination IP addresses, ports, IP length (from the IP header), and APDU length (from the IEC 104 header). Additionally, features such as the APDU format type, u-format type (including start data transfer, stop data transfer, test frame activation, test frame confirmation, stop data transfer action, and stop data transfer confirmation), and ASDU type identification (e.g., single point information, interrogation command) are recorded. These headers provide detailed information on the ASDU packet, including the number of Information Objects within it, the cause of transmission (e.g., periodic, spontaneous), originator address, ASDU address field, and Information Object Address—a list of addresses of Information Objects present in the ASDU. It covers a range of benign and malicious activities, including denial-of-service (DoS) attacks, injection attacks, connloss attacks, and rogue device attacks. For further details about the BUT-IEC104-I dataset, see [47,48].

Each instance in the dataset consists of a sequence of network packets exchanged between the different components of the smart grid, such as substations and control centers, encoded according to the IEC 104 protocol standards. The packets contain information about the command and data exchanges, status updates, and control signals, reflecting the operational activities typical in power grid systems. To facilitate the evaluation of the anomaly detection methods, the dataset is annotated with labels indicating the presence or absence of malicious activity in each network traffic instance. This annotation enables researchers to assess the performance of the anomaly detection algorithms in accurately distinguishing between normal and malicious behaviors. Furthermore, the dataset includes metadata providing additional contextual information, such as timestamps, source and destination addresses, packet sizes, and communication patterns. This metadata enhances the understanding of the underlying network dynamics and aids in analyzing the detected anomalies. The dataset contains not only normal traffic but also various types of attack events, including DOS attacks, switching attacks, injection command attacks, and connection loss attacks.

All attacks were intentionally generated at the L7 layer, also known as the application layer, of the OSI (Open Systems Interconnection) model. The application layer is the topmost layer in the OSI model and provides network services directly to user applications. Attacks at this layer can exploit the vulnerabilities in the specific applications or protocols running on top of the network stack. By generating attacks at the application layer, the aim was to simulate realistic cyber threats that target smart grid networks. These attacks include various forms of malicious activities, such as denial-of-service (DoS) attacks, injection attacks, command blocking, switching attacks, and other network intrusions. Attacking the application layer allowed for the assessment of the effectiveness of the anomaly detection methods in detecting and mitigating these threats. It also enabled the evaluation of the robustness of the detection framework in identifying the anomalies within the application-level traffic of smart grid networks. The attack traces were generated during the Bonnet research project from 2019 to 2022. More details are available at https://www.fit.vut.cz/research/project/1303/ (accessed on 20 May 2024).

Overall, the BUT-IEC104-I dataset serves as a comprehensive and realistic benchmark for evaluating the efficacy of the anomaly detection techniques in safeguarding smart grid infrastructures against cyber threats. Its diverse composition and detailed annotations make it a valuable resource for advancing the research in power grid cybersecurity.

4.2. Detection Results

The anomaly detection methods were trained using normal IEC 104 communication data (attack-free data), which consisted of 58,930 packets collected over a period of 2 days, 19 h, and 55 min. This approach was chosen because the proposed methods are semi-supervised, eliminating the need for labeled data. From each traffic sample, 16 key attributes were extracted, including source and destination IP addresses, as well as port numbers. During the training, the parameters of each method were adjusted to minimize the cross-entropy of the reconstructed error, ensuring the optimal performance of the models.

The AE-GRU model is configured with nine input features and utilizes the ReLU activation and Cross-Entropy loss functions. The Rmsprop optimizer was employed during training, which consists of 300 epochs with a batch size of 250. Each input sequence has a length of 12 timesteps. The encoder comprises three layers, with the first layer containing a GRU with 128 units, followed by a GRU layer with 16 units, and finally, a dense layer with units equal to the number of features. On the other hand, the decoder comprises three layers: the first layer is a dense layer with the units equal to the number of features, followed by two GRU layers with 16 and 128 units, respectively. On the other hand, the standalone anomaly detection methods are configured with the specific parameters for their operation. Isolation Forest is set with a contamination rate of 0.01 and utilizes 150 estimators. The LOF method also has a contamination rate of 0.01 and is configured with the novelty set to TRUE. The EE method is configured with a contamination rate of 0.01 and a support fraction of 0.995. For the 1SVM method, the kernel used is the Radial Basis Function (RBF), with the parameters set as follows: $\nu = 0.0015$ and $\gamma = 0.25$.

- DOS attack: In this scenario, a DOS attack is directed at an IEC 104 control station, aiming to overload it and disrupt its functionality, thereby potentially causing grid failure. The attacker gains unauthorized access by utilizing a spoofed IP address and floods the victim with a significant volume of 1049 messages over a 30 min period. This onslaught overwhelms the control station's resources, rendering it unable to respond effectively to legitimate requests and ultimately leading to system instability or failure.

Figure 5 presents the heatmap illustrating the detection performance metrics attained by the AE-GRU-based and standalone anomaly detection methods during the DOS attack scenario. During the DOS attack scenario, Figure 5 shows that the AE-GRU-based anomaly detection methods demonstrated varying performance levels compared to the standalone models. Specifically, the AE-GRU-iForest method achieved an accuracy of 0.8875, exhibiting strong precision and F1-score values of 1 and 0.9052, respectively. This suggests that the AE-GRU-iForest method effectively detected anomalies with minimal false positives and achieved a balance between precision and recall. However, the AE-GRU-LOF method showed a lower accuracy of 0.65, indicating that it struggled to correctly classify instances during the attack. While it achieved perfect recall, its precision and F1-score were notably lower at 0.65 and 0.7879, respectively. Conversely, both the AE-GRU-1SVM and AE-GRU-EE methods demonstrated excellent performance, with accuracies exceeding 0.99 and perfect precision, recall, and F1-scores, indicating their robustness in detecting DOS attacks.

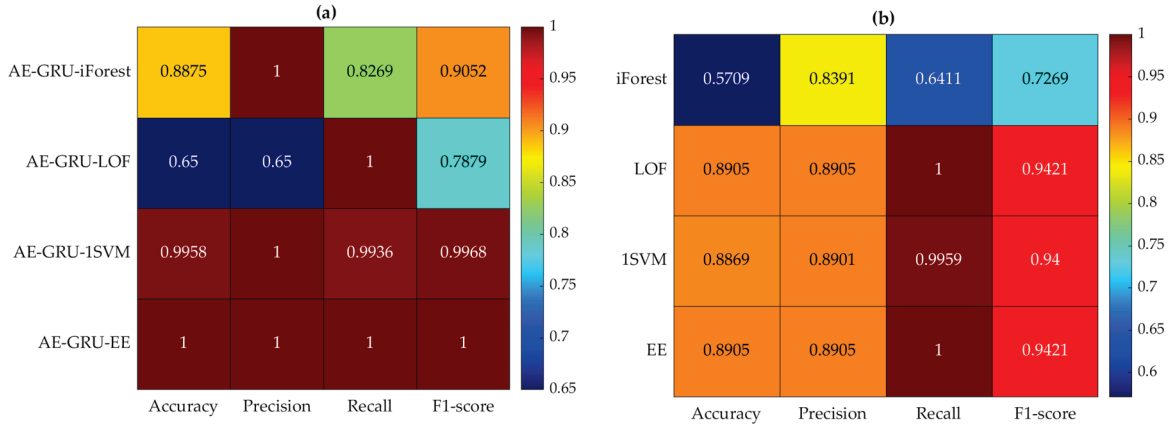


Figure 5. Detection performance comparison of (**a**) the AE-GRU-based and (**b**) standalone anomaly detection methods during the DOS attack scenario.

In comparison, the standalone anomaly detection methods exhibited different detection capabilities (Figure 5b). The iForest method achieved an accuracy of 0.5709, indicating a lower overall performance compared to the AE-GRU-based methods. Although it exhibited reasonable precision and F1-score values of 0.8391 and 0.7269, respectively, its recall was notably lower at 0.6411, indicating a higher rate of false negatives. On the other hand, the LOF and EE methods achieved an accuracy of 0.8905, with perfect precision, recall, and F1-scores, suggesting their effectiveness in detecting DOS attacks. However, the 1SVM method displayed a slightly lower accuracy of 0.8869, with precision, recall, and F1-score values of 0.8901, 0.9959, and 0.94, respectively, indicating its high recall but lower precision compared to the other methods.

Overall, the AE-GRU-based anomaly detection methods, particularly the AE-GRU-iForest and AE-GRU-1SVM methods, outperformed the standalone models in detecting DOS attacks, achieving higher accuracies and balanced precision–recall trade-offs. This highlights the effectiveness of leveraging deep learning techniques, such as the AE-GRU, for anomaly detection in smart grid cybersecurity applications.

- Switching attack: In a switching attack, malware is utilized to manipulate the targeted devices, turning them alternately on and off. This type of attack disrupts the normal operation of the devices and can potentially lead to system instability or malfunction. In this scenario, the switching attack persisted for 10 min, during which 72 packets were transmitted to the victim station. The attacker's objective is to disrupt the station's functionality by manipulating its operational status, which can have significant repercussions on the overall grid operation and stability. Figure 6 displays a heatmap comparing the detection performance metrics of the AE-GRU-based and standalone anomaly detection methods during the switching attack scenario. In the switching attack scenario, both the AE-GRU-based and standalone anomaly detection methods were evaluated. The AE-GRU-based methods, like the iForest and LOF methods, showed good accuracy and precision, with the iForest method achieving 0.8903 accuracy and 0.9856 precision, and the LOF method achieving 0.8945 accuracy and good precision (Figure 6a). However, the 1SVM model performed poorly with only 0.1181 accuracy and 0.0833 recall. The EE model had moderate accuracy at 0.8692 and precision at 0.9852. The standalone methods, especially the LOF and EE methods, demonstrated a strong performance, with accuracies of 0.9576 and perfect precision scores (Figure 6b). The 1SVM model also had high accuracy and precision, with scores of 0.9556 and 0.9575, respectively. Overall, while the AE-GRU-based methods like the iForest and LOF methods performed well, the standalone methods, particularly the LOF and EE methods, consistently showed a strong performance, suggesting their effectiveness in detecting the anomalies in the switching attack scenario.

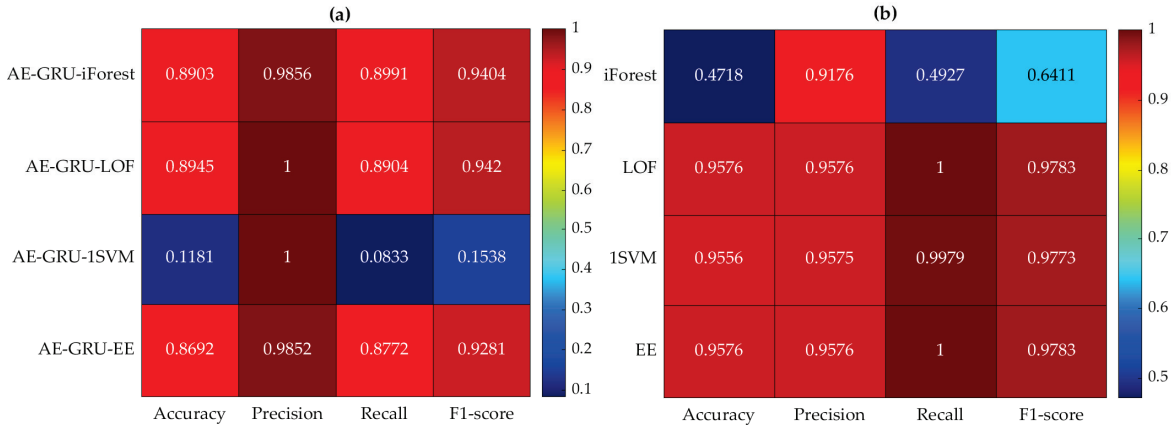


Figure 6. Detection performance comparison of (**a**) the AE-GRU-based and (**b**) standalone anomaly detection methods during the switching attack scenario.

- Injection command attacks: In injection command attacks, the attacker interferes with a connected device by modifying its configurations or introducing fraudulent commands, leading to abnormal behavior. This manipulation results in the generation of various anomalies within the system. The attack occurred in two distinct instances: the first lasted for 5 min and involved 83 packets, while the second persisted for 15 min and comprised 221 packets. The attacker aims to compromise the integrity and functionality of the targeted device, potentially causing disruptions or damage to the overall system. Such attacks pose significant risks to the security and reliability of the network, highlighting the importance of robust cybersecurity measures in safeguarding critical infrastructure like smart grids. Figure 7 illustrates the heatmap comparing the detection performance metrics of the AE-GRU-based and standalone anomaly detection methods during the injection attack scenario based on the testing data. In the injection attack scenario, both the AE-GRU-based and standalone anomaly detection methods achieved high accuracy, precision, and recall. The AE-GRU-based methods had accuracy values from 0.9619 to 0.9771, while the standalone methods ranged from 0.457 to 0.9275. All the methods had precision scores of 1, indicating near-perfect identification of attack instances. The AE-GRU-based methods showed recall values from 0.9616 to 1, while the standalone methods ranged from 0.4927 to 1. Overall, the AE-GRU-based methods consistently outperformed the standalone methods, demonstrating their superior detection capabilities for injection attacks.

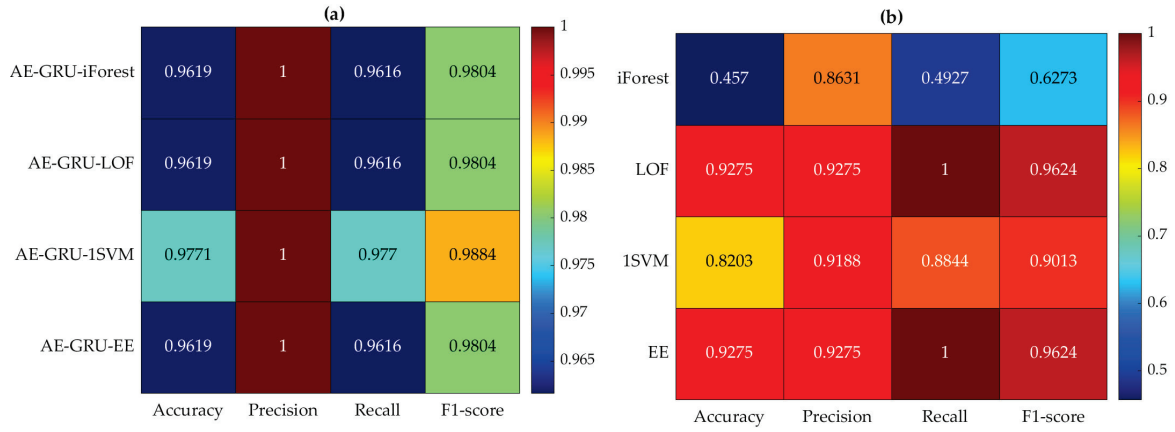


Figure 7. Detection performance comparison of (**a**) the AE-GRU-based and (**b**) standalone anomaly detection methods during the injection attack scenario.

- Connection loss attacks: In this category, the attacker attempts to interfere with specific devices, disrupting their interconnected communications. Two scenarios were simulated to replicate this interference. In the first scenario, a 10 min disconnection led to the loss of 146 packets. In the second scenario, the disconnection persisted for an hour, losing several packets. These attacks aim to disrupt the normal functioning of the targeted devices, potentially causing delays, malfunctions, or data loss. Such interference poses significant threats to the reliability and integrity of the communication network, underscoring the importance of robust cybersecurity measures in mitigating such risks.

Figure 8 shows a heatmap comparing the evaluation metrics of the AE-GRU and standalone anomaly detection methods during a connection loss attack scenario, based on the test data. From Figure 8, we observe that the AE-GRU-based methods, specifically the AE-GRU-LOF and AE-GRU-EE methods, exhibited a significantly higher detection performance compared to the standalone methods, with F1-score values of 94.93% and 96.43%, respectively. This signifies a balanced performance in precision and recall, crucial for accurate anomaly detection. Conversely, the standalone methods, including the iForest and 1SVM methods, demonstrated lower F1-scores, with the iForest method reaching 51.37% and the 1SVM method reaching 83.29%. This suggests that while the standalone methods might excel in certain aspects such as recall, their overall performance in terms of precision and recall balance is not as robust as the AE-GRU-based methods are. These differences in F1-score highlight the effectiveness of the AE-GRU-based methods in achieving a harmonious trade-off between precision and recall, thereby offering a more balanced approach to the anomaly detection during connection loss attacks. This balance is crucial in ensuring the accurate detection of anomalies while minimizing false positives, ultimately enhancing the cybersecurity of smart grid networks.

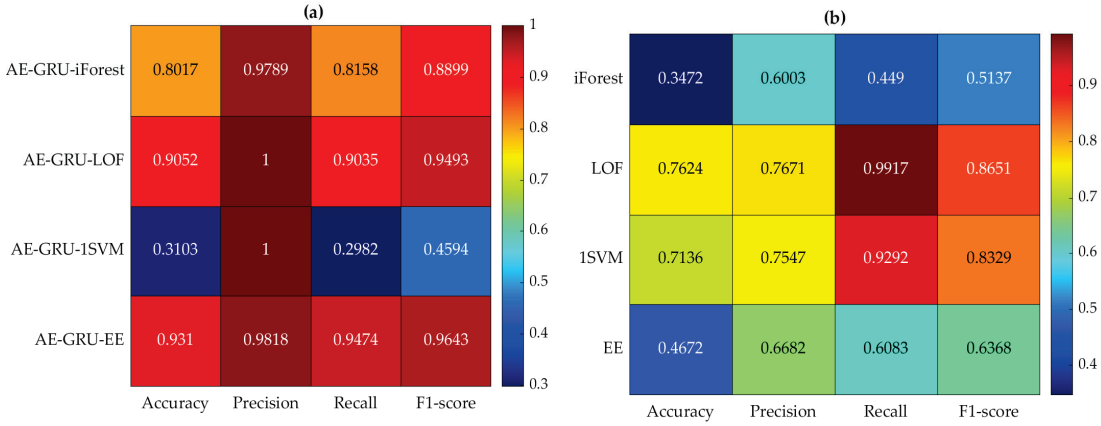


Figure 8. Detection performance comparison of (**a**) the AE-GRU-based and (**b**) standalone anomaly detection methods during the connection loss attack scenario.

- Rogue device attack: In this attack, the attacker gains unauthorized access to the communication network, allowing them to transmit arbitrary messages to authorized devices. These messages compel the devices to perform unauthorized activities, leading to irregular events and potentially harmful outcomes. In a 30 min attack, the attacker transmitted 417 abnormal messages, exploiting the network security protocol vulnerabilities. Such attacks can disrupt the network's normal operations, compromise its integrity, and pose significant risks to the overall cybersecurity posture of the system. It underscores the importance of robust security measures to prevent unauthorized access and mitigate the impact of such attacks on the network's functionality and reliability.

In the rogue device attack scenario, the AE-GRU-based anomaly detection methods outperformed the standalone methods (Figure 9a,b). The AE-GRU-based methods, such as

the iForest and LOF methods, achieved impressive F1-scores of 99.03%, indicating highly accurate detection with minimal false positives and negatives. On the other hand, the standalone methods showed a more varied performance. While the LOF method achieved a perfect F1-score, the iForest method exhibited a lower F1-score of 60.04%, indicating a less balanced detection performance. Figure 9b reveals that the 1SVM method achieved an F1-score of 86.95%, demonstrating relatively balanced precision and recall compared to the AE-GRU-based methods. However, it still fell short of the performance attained by the AE-GRU-based methods. These quantitative results highlight the superior performance of the AE-GRU-based anomaly detection methods in accurately identifying the rogue devices within smart grid networks. Leveraging deep learning and sequential data modeling, these methods offer a robust approach to enhancing the cybersecurity by effectively detecting anomalous behavior.

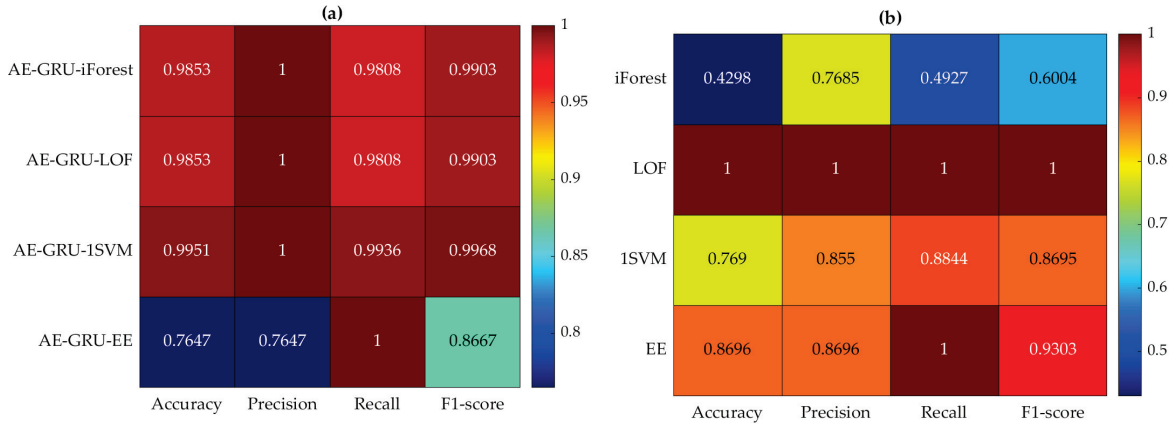


Figure 9. Detection performance comparison of (**a**) the AE-GRU-based and (**b**) standalone anomaly detection methods during the rogue device attack scenario.

While no single approach exhibited a superior performance across all the considered attacks, the AE-GRU-based anomaly detection methods consistently outperformed the standalone anomaly detection methods (LOF, EE, 1SVM, iForest) in detecting the various types of attacks in smart grid networks. This superiority can be attributed to several factors. Firstly, the AE-GRU-based methods leverage the power of deep learning architectures, specifically autoencoders and GRUs, to effectively model the complex temporal dependencies inherent in time-series data such as network traffic. This enables them to capture the subtle patterns and anomalies that may be missed by the standalone methods. Additionally, the AE-GRU models are capable of learning the high-level representations of data features through unsupervised learning, allowing them to create more informative and discriminative feature representations compared to the standalone methods.

Figure 10 depicts the average performance metrics of the anomaly detection methods considered. The benefit of evaluating the average performance metrics is providing a comprehensive assessment of each method's overall effectiveness across different types of attacks. By averaging the results over multiple scenarios, we gain insights into the methods' robustness and adaptability in detecting the anomalies commonly encountered in smart grid networks. We observe that the AE-GRU-based EE and LOF methods stand out as the top performers, achieving high accuracy and F1-scores of 0.91 and 0.95, respectively. These methods also demonstrate superior precision and recall, indicating their ability to effectively identify anomalies while minimizing false positives and false negatives. Furthermore, Figure 10 underscores the advantage of leveraging advanced techniques such as the AE-GRU-EE, which combines autoencoder-based feature extraction with a semi-supervised anomaly detection method, resulting in a superior detection performance. Similarly, the AE-GRU-LOF method, a density-based outlier detection method, proves highly effective in anomaly detection tasks.

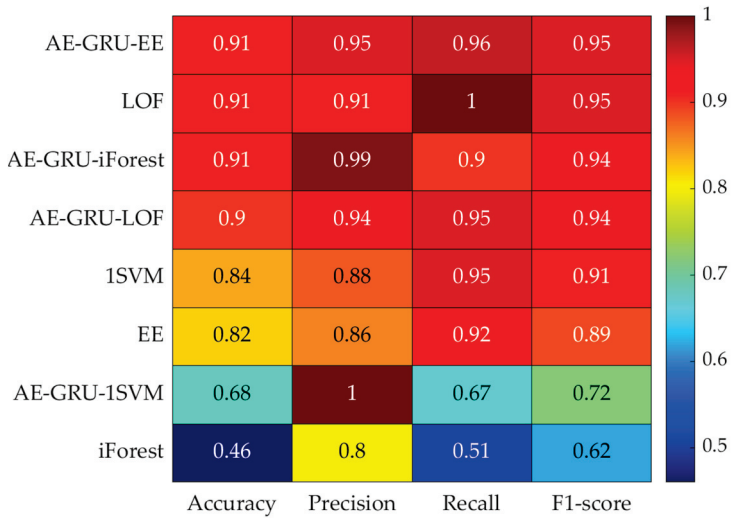


Figure 10. Average performance metrics of the considered anomaly detection methods.

5. Discussion and Conclusion

Protecting smart grids from cyberattacks is paramount in regard to ensuring the reliability and safety of grid operations. By deploying robust detection mechanisms and response strategies, operators can mitigate the impact of cyber threats and safeguard the grid infrastructure. The study demonstrates the effectiveness of the AE-GRU-based anomaly detection methods in enhancing the cybersecurity of smart grid networks using the BUT-IEC104-I dataset. The advantages of this approach include its ability to accurately detect the various types of anomalies, such as DOS attacks, switching attacks, injection command attacks, and connection loss attacks. The integration of the AE-GRU models with the semi-supervised anomaly detection methods, such as the EE and LOF methods, enhances the adaptability and robustness, resulting in an improved detection performance across the different attack scenarios. This highlights our framework's efficacy and the benefits of integrating the AE-GRU with semi-supervised anomaly detection to identify the cyber threats in smart grid traffic.

Despite the satisfactory detection performance, the study also has limitations that warrant attention. One limitation is the reliance on simulated attack scenarios, which may not fully capture the complexity and variability of real-world cyber threats. Additionally, the performance of the proposed framework may be influenced by factors such as the quality and quantity of training data, as well as the specific configurations of the smart grid network under consideration. Several potential solutions can be considered to enhance the performance of the proposed framework. Firstly, increasing the diversity and authenticity of the training data by incorporating real-world network traffic traces can improve the robustness of the AE-GRU models.

The encouraging outcomes (performance) of the presented cyberattack detection solution motivate further exploration and potential integration into the monitoring and intrusion detection systems across diverse industrial domains. Moreover, future investigations will be conducted by incorporating wavelet techniques to handle noisy data. Additionally, we intend to investigate the effectiveness of joining (merging) unsupervised deep learning with statistical monitoring charts, such as the Generalized Likelihood Ratio Test. Another potential line of improvement is to explore techniques such as transfer learning or domain adaptation that could help leverage the knowledge from the related domains to improve the generalization and adaptability of the AE-GRU models to diverse smart grid environments. Moreover, incorporating the ensemble learning approaches, where multiple models are combined to make predictions, could potentially improve the overall robustness and reliability of the anomaly detection system.

Author Contributions: F.H.: Writing—Review and editing, Formulation, Methodology, Investigation; Validation; B.B.: Writing, Review and editing, Conceptualization, Validation; A.D.: Writing—Review and editing, Methodology, Software, Formal analysis, Validation; Y.S.: Review and Editing, Formal analysis, Supervision, Visualization. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used in this study are available at https://ieee-dataport.org/documents/ics-dataset-smart-grid-anomaly-detection (accessed on 20 May 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Tan, S.; De, D.; Song, W.Z.; Yang, J.; Das, S.K. Survey of security advances in smart grid: A data driven approach. *IEEE Commun. Surv. Tutor.* **2016**, *19*, 397–422. [CrossRef]
2. Khoei, T.T.; Slimane, H.O.; Kaabouch, N. A comprehensive survey on the cyber-security of smart grids: Cyber-attacks, detection, countermeasure techniques, and future directions. *arXiv* **2022**, arXiv:2207.07738.
3. Goudarzi, A.; Ghayoor, F.; Waseem, M.; Fahad, S.; Traore, I. A survey on IoT-enabled smart grids: Emerging, applications, challenges, and outlook. *Energies* **2022**, *15*, 6984. [CrossRef]
4. Tang, D.; Fang, Y.P.; Zio, E. Vulnerability analysis of demand-response with renewable energy integration in smart grids to cyber attacks and online detection methods. *Reliab. Eng. Syst. Saf.* **2023**, *235*, 109212. [CrossRef]
5. Ding, J.; Qammar, A.; Zhang, Z.; Karim, A.; Ning, H. Cyber threats to smart grids: Review, taxonomy, potential solutions, and future directions. *Energies* **2022**, *15*, 6799. [CrossRef]
6. Kim, Y.; Hakak, S.; Ghorbani, A. Smart grid security: Attacks and defence techniques. *IET Smart Grid* **2023**, *6*, 103–123. [CrossRef]
7. Faquir, D.; Chouliaras, N.; Sofia, V.; Olga, K.; Maglaras, L. Cybersecurity in smart grids, challenges and solutions. *AIMS Electron. Electr. Eng.* **2021**, *5*, 24–37.
8. Sahani, N.; Zhu, R.; Cho, J.H.; Liu, C.C. Machine learning-based intrusion detection for smart grid computing: A survey. *ACM Trans. Cyber-Phys. Syst.* **2023**, *7*, 1–31. [CrossRef]
9. An, D.; Yang, Q.; Liu, W.; Zhang, Y. Defending against data integrity attacks in smart grid: A deep reinforcement learning-based approach. *IEEE Access* **2019**, *7*, 110835–110845. [CrossRef]
10. Sakhnini, J.; Karimipour, H.; Dehghantanha, A. Smart grid cyber attacks detection using supervised learning and heuristic feature selection. In Proceedings of the 2019 IEEE 7th International Conference on Smart Energy Grid Engineering (SEGE), Oshawa, ON, Canada, 12–14 August 2019; pp. 108–112.
11. Hink, R.C.B.; Beaver, J.M.; Buckner, M.A.; Morris, T.; Adhikari, U.; Pan, S. Machine learning for power system disturbance and cyber-attack discrimination. In Proceedings of the 2014 7th International Symposium on Resilient Control Systems (ISRCS) 2014, Denver, CO, USA, 19–21 August 2014; pp. 1–8.
12. Esmalifalak, M.; Liu, L.; Nguyen, N.; Zheng, R.; Han, Z. Detecting stealthy false data injection using machine learning in smart grid. *IEEE Syst. J.* **2014**, *11*, 1644–1652. [CrossRef]
13. Ahmed, S.; Lee, Y.; Hyun, S.H.; Koo, I. Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forest. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2765–2777. [CrossRef]
14. Acosta, M.R.C.; Ahmed, S.; Garcia, C.E.; Koo, I. Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks. *IEEE Access* **2020**, *8*, 19921–19933. [CrossRef]
15. Al-Qudah, M.; Ashi, Z.; Alnabhan, M.; Abu Al-Haija, Q. Effective one-class classifier model for memory dump malware detection. *J. Sens. Actuator Netw.* **2023**, *12*, 5. [CrossRef]
16. Menon, D.M.; Radhika, N. Anomaly detection in smart grid traffic data for home area network. In Proceedings of the 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, India, 18–19 March 2016; pp. 1–4.
17. Al-Abassi, A.; Karimipour, H.; Dehghantanha, A.; Parizi, R.M. An ensemble deep learning-based cyber-attack detection in industrial control system. *IEEE Access* **2020**, *8*, 83965–83973. [CrossRef]
18. Kundu, A.; Sahu, A.; Serpedin, E.; Davis, K. A3D: Attention-based auto-encoder anomaly detector for false data injection attacks. *Electr. Power Syst. Res.* **2020**, *189*, 106795. [CrossRef]
19. Tuor, A.; Kaplan, S.; Hutchinson, B.; Nichols, N.; Robinson, S. Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. In Proceedings of the Workshops at the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
20. Haghshenas, S.H.; Hasnat, M.A.; Naeini, M. A temporal graph neural network for cyber attack detection and localization in smart grids. In Proceedings of the 2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 16–19 January 2023; pp. 1–5.
21. Khanna, K.; Panigrahi, B.K.; Joshi, A. AI-based approach to identify compromised meters in data integrity attacks on smart grid. *IET Gener. Transm. Distrib.* **2018**, *12*, 1052–1066. [CrossRef]

22. Karimipour, H.; Dehghantanha, A.; Parizi, R.M.; Choo, K.K.R.; Leung, H. A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids. *IEEE Access* **2019**, *7*, 80778–80788. [CrossRef]
23. Qiu, W.; Tang, Q.; Wang, Y.; Zhan, L.; Liu, Y.; Yao, W. Multi-view convolutional neural network for data spoofing cyber-attack detection in distribution synchrophasors. *IEEE Trans. Smart Grid* **2020**, *11*, 3457–3468. [CrossRef]
24. Mohammadpourfard, M.; Genc, I.; Lakshminarayana, S.; Konstantinou, C. Attack detection and localization in smart grid with image-based deep learning. In Proceedings of the 2021 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Aachen, Germany, 25–28 October 2021; pp. 121–126.
25. Ali, S.; Li, Y. Learning multilevel auto-encoders for DDoS attack detection in smart grid network. *IEEE Access* **2019**, *7*, 108647–108659. [CrossRef]
26. Dou, C.; Wu, D.; Yue, D.; Jin, B.; Xu, S. A hybrid method for false data injection attack detection in smart grid based on variational mode decomposition and OS-ELM. *CSEE J. Power Energy Syst.* **2020**, *8*, 1697–1707.
27. Lu, G.; Tian, X. An efficient communication intrusion detection scheme in AMI combining feature dimensionality reduction and improved LSTM. *Secur. Commun. Netw.* **2021**, *2021*, 6631075. [CrossRef]
28. Qi, R.; Rasband, C.; Zheng, J.; Longoria, R. Detecting cyber attacks in smart grids using semi-supervised anomaly detection and deep representation learning. *Information* **2021**, *12*, 328. [CrossRef]
29. Ferrag, M.A.; Maglaras, L. DeepCoin: A novel deep learning and blockchain-based energy exchange framework for smart grids. *IEEE Trans. Eng. Manag.* **2019**, *67*, 1285–1297. [CrossRef]
30. Liu, W.; Wang, Z.; Liu, X.; Zeng, N.; Liu, Y.; Alsaadi, F.E. A survey of deep neural network architectures and their applications. *Neurocomputing* **2017**, *234*, 11–26. [CrossRef]
31. Zhang, G.; Liu, Y.; Jin, X. A survey of autoencoder-based recommender systems. *Front. Comput. Sci.* **2020**, *14*, 430–450. [CrossRef]
32. Michelucci, U. An introduction to autoencoders. *arXiv* **2022**, arXiv:2201.03898.
33. Dey, R.; Salem, F.M. Gate-variants of gated recurrent unit (GRU) neural networks. In Proceedings of the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), Boston, MA, USA, 6–9 August 2017; pp. 1597–1600.
34. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
35. Kanai, S.; Fujiwara, Y.; Iwamura, S. Preventing gradient explosions in gated recurrent units. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–10.
36. Ravanelli, M.; Brakel, P.; Omologo, M.; Bengio, Y. Improving speech recognition by revising gated recurrent units. *arXiv* **2017**, arXiv:1710.00641.
37. Harrou, F.; Taghezouit, B.; Bouyeddou, B.; Sun, Y. Cybersecurity of photovoltaic systems: Challenges, threats, and mitigation strategies: A short survey. *Front. Energy Res.* **2023**, *11*, 1274451. [CrossRef]
38. Rabaoui, A.; Davy, M.; Rossignol, S.; Lachiri, Z.; Ellouze, N. Improved one-class SVM classifier for sounds classification. In Proceedings of the 2007 IEEE Conference on Advanced Video and Signal Based Surveillance, London, UK, 5–7 September 2007; pp. 117–122.
39. Kawi, O.; Clawson, K.; Dunn, P.; Knight, D.; Hodgson, J.; Peng, Y. Medical formulation recognition (MFR) using deep feature learning and one class SVM. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–7.
40. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; pp. 413–422.
41. McKinnon, C.; Carroll, J.; McDonald, A.; Koukoura, S.; Infield, D.; Soraghan, C. Comparison of new anomaly detection technique for wind turbine condition monitoring using gearbox SCADA data. *Energies* **2020**, *13*, 5152. [CrossRef]
42. Hubert, M.; Debruyne, M.; Rousseeuw, P.J. Minimum covariance determinant and extensions. *Wiley Interdiscip. Rev. Comput. Stat.* **2018**, *10*, e1421. [CrossRef]
43. Alghushairy, O.; Alsini, R.; Soule, T.; Ma, X. A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data Cogn. Comput.* **2020**, *5*, 1. [CrossRef]
44. Lin, C.Y.; Nadjm-Tehrani, S. Understanding IEC-60870-5-104 traffic patterns in SCADA networks. In Proceedings of the 4th ACM Workshop on Cyber-Physical System Security, Incheon, Republic of Korea, 4 June 2018; pp. 51–60.
45. Cherifi, T.; Hamami, L. A practical implementation of unconditional security for the IEC 60780-5-101 SCADA protocol. *Int. J. Crit. Infrastruct. Prot.* **2018**, *20*, 68–84. [CrossRef]
46. Grammatikis, P.R.; Sarigiannidis, P.; Sarigiannidis, A.; Margounakis, D.; Tsiakalos, A.; Efstathopoulos, G. An anomaly detection mechanism for IEC 60870-5-104. In Proceedings of the 2020 9th International Conference on Modern Circuits and Systems Technologies (MOCAST), Bremen, Germany, 7–9 September 2020; pp. 1–4.

47. Matoušek, P.; Ryšavý, O.; Grégr, M.; Havlena, V. Flow based monitoring of ICS communication in the smart grid. *J. Inf. Secur. Appl.* **2020**, *54*, 102535.
48. Matoušek, P.; Havlena, V.; Holík, L. Efficient modelling of ICS communication for anomaly detection using probabilistic automata. In Proceedings of the 2021 IFIP/IEEE International Symposium on Integrated Network Management (IM), Bordeaux, France, 17–21 May 2021; pp. 81–89.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Article

A Secure Auditable Remote Registry Pattern for IoT Systems

Antonio Maña *, Francisco J. Jaime and Lucía Gutiérrez

Institute for Software Engineering and Software Technology "Jose María Troya Linero", Universidad de Málaga, 29071 Málaga, Spain; franj@uma.es (F.J.J.); luciagutierrezmolina@uma.es (L.G.)
* Correspondence: amana@uma.es

Abstract: In software engineering, pattern papers serve the purpose of providing a description of a generalized, reusable solution to recurring design problems, based on practical experience and established best practices. This paper presents an architectural pattern for a Secure Auditable Registry service based on Message-Oriented Middleware to be used in large-scale IoT systems that must provide auditing capabilities to external entities. To prepare the pattern, the direct experience in applying the pattern solution in an industry-funded R&D project has been a key aspect because it has allowed us to gain a deep understanding of the problem and the solution, and it has contributed to the correctness and real-world applicability of the pattern as described. To further improve the quality of the paper, we have followed the commonly accepted practices in pattern development (including peer reviews) to ensure that the core aspects of the solution are correctly represented and that the description allows it to be applicable to similar problems in other domains, such as healthcare, autonomous devices, banking, food tracing or manufacturing to name a few. The work done in applying this pattern confirms that it solves a recurring problem for IoT systems, but also that it can be adopted in other domains, providing an effective solution in order to achieve enhancement of the auditability capabilities of the target systems. This pattern will be part of a pattern language (i.e., a family of related patterns) that we are developing for transitioning from legacy systems to IoT with an emphasis on security.

Keywords: IoT; security architecture; software security patterns; pattern language; design patterns; security patterns

1. Introduction and Methodology

1.1. Introduction

The global trend towards connected IT systems and services capable of providing highly customizable information and analytics in near-real time requires fast and intelligent applications using new architectures and technologies. In this context, many legacy systems cannot meet the stringent requirements set by IoT applications or respond to the explosive growth of data generated by connected devices (IoT, IIoT (Industrial IoT) and OT (Operational Technology) are different abbreviations that are often used in combination. OT (sometimes used as a synonym for IIoT) is more industry-focused and refers to the hardware and software used to identify, monitor and control physical devices, processes and events in a manufacturing environment). The advent of the Internet of Things has been such a game changer that it has become one of the biggest drivers in the market to retire these legacy systems. In 2021, the value of the global IoT market was estimated at USD 213 billion, and forecasts suggest that it will reach between USD 2400 and USD 8100 billion by 2030 [1,2], with the number of IoT devices expected to potentially exceed 29 billion in a variety of industrial and consumer applications [3].

In the light of these numbers, and the corroboration of how fast legacy systems are becoming unfit for their purposes in the current IT context, it is clear that the decision to re-engineer legacy applications that are no longer able to meet their business objectives cannot be put off. Indeed, this situation is rather frequent in other domains involving

hardware devices, such as banking, healthcare, manufacturing, food-chain tracking and security [4].

We believe that the criticality and importance of this transition and the large number of systems that could be affected by such a process justifies the need to provide tools to facilitate and support the process of re-engineering and adapting legacy systems to new technologies such as IoT, microservices, etc. In fact, this need has already been recognized for other technological changes, and we can find, for instance, patterns for replacing legacy systems, for moving to the cloud [5,6] or even for developing systems [7].

Patterns are one of the most useful tools in software engineering in general and in re-engineering in particular. Patterns capture reusable solutions to common design and architectural problems. Patterns cover a wide range of problems and solutions on different levels, such as design (e.g., Singleton, Observer) and architectural levels (e.g., MVC, microservices), providing proven templates that developers can apply to solving problems while achieving specific properties like quality, maintainability and scalability. Patterns promote consistency, reduce complexity and improve collaboration across teams by providing a shared vocabulary for addressing recurring issues. Ultimately, patterns help streamline development processes, leading to more robust and flexible software systems.

The scientific contribution of this paper is the formulation of a new architectural pattern that is a software engineering tool to help other practitioners to better understand the problem, and to describe for them a specific proven and reusable solution to it. To produce this contribution, our work involved investigating in depth the solution, both in theoretical terms and through first-hand experience in a real industry-funded project, and extracting the essence of the problem and the solution in a form that would (1) help other practitioners and researchers learn the main aspects that are important in order to decide whether the solution is applicable to their system, and (2) facilitate the application of the solution to other systems, which, of course, requires research, generalization and polishing of the description (e.g., by removing implementation details that depend on specific technologies).

This paper presents an architectural pattern for a secure auditable registry service to be used by systems that require audit from external entities. This is a common requirement in IoT systems but is also relevant for many other distributed systems. The pattern has been crafted based on direct experience in solving the target problem in an R&D project focused on the re-engineering of a legacy large-scale high-security system in the domain of gambling, but we have followed the commonly accepted practices in pattern development (including peer reviews) to ensure that the core aspects of the solution are correctly represented and that the description allows it to be applicable to similar problems in other domains, such as healthcare, autonomous devices, banking, food tracing or manufacturing, to name a few.

To avoid confusion, in this paper we will refer to "systems" when we talk about general ones and will use the acronym SUD (System Under Design) to refer to a specific system that is being designed/developed.

The structure of the remainder of the paper is as follows: Section 2 introduces the context and background, and it presents the pattern ecosystem of the SARMOM pattern, which includes other patterns that may be used together with it, depending on the requirements of the SUD. Section 3 shows the pattern ecosystem around SARMOM. Section 4 contains the proposed pattern, described using a security-oriented variant of the POSA template, which is the one most frequently used in patterns dealing with security aspects [8]. Finally, Section 5 describes the conclusions and future work.

1.2. Methodology

Due to their practical and empirical nature, in the definition of patterns a typical scientific approach with hypotheses and experiments is not suitable, but this does not mean that this process lacks a methodology. There are well-established principles and practices to guarantee the quality of the pattern description, which have been perfected

by the pattern community for a long time (the PLoP conference held its 30th edition this year). In particular, it is common in the production of patterns to count on peer reviews and to produce several iterations of the pattern description. In the preparation of the pattern presented in Section 4, we have followed the methodology developed by the Hillside Group for the Pattern Languages of Programs (PLoP) conferences, which is centered on collaborative exploration and refinement of design solutions through a structured process. This process typically begins with the identification of a recurring problem in software design and the creation of initial pattern drafts. This step is taken during the development of an industry-funded project, and it includes a process of analysis of the solution in other existing real-world systems. Then, the drafts undergo peer review rounds, where authors present their patterns in small groups (we did this both internally, with members of our client staff, and externally, with the participation of experts in the definition of patterns). Feedback from these sessions allows for iterative improvements and helps authors articulate their patterns more clearly. This methodology emphasizes real-world applicability and collective insights, with a focus on refining patterns through discussions that highlight their context, forces and consequences. The Hillside Group's methodology is grounded in the belief that patterns are best understood and improved in a community setting, fostering a shared understanding of design challenges and solutions. This iterative, community-driven process has led to the establishment of a rich body of knowledge in software patterns, influencing both academic research and practical software development.

2. Motivation and Background

2.1. Motivating Scenario

Modern IT systems, and especially IoT systems, must comply with different standards and regulations, and they are expected to fulfill essential non-functional (a.k.a. quality) requirements, such as reliability and security. In particular, accountability, auditability and transparency are becoming essential security properties in these systems.

The commonly accepted definition of quality includes different aspects: efficiency, operability, compatibility, security or maintainability among others. We can refer to the ISO/IEC 25010:2011 software quality model [9] for an in-depth description of IT systems quality. In our case, the motivating scenario is a project for re-engineering a system to control slot machines in a casino. In this context, evolvability and maintainability are essential because the SUD must remain in operation for a long time and adapt to different changes (the current communication protocol for slot machines will soon be replaced, new functionalities are constantly added, etc.). To this end, the architecture proposed by the Secure Auditable Registry based on Message-Oriented Middleware (SARMOM) pattern is ideal, in order to achieve those properties and to reduce and isolate the impact of the future changes in the system.

Another critical requirement (which is also part of quality [9]) is security. Security properties cover an extremely wide spectrum of targets and goals ranging from the classic CIA trio (Confidentiality, Integrity and Availability) to more complex and specific properties, such as non-repudiation, access control mechanisms and policies, accountability, auditability, freshness, retention, etc. Together with the fact that there are agents interested in breaking those properties, the variety and complexity of the technical security solutions make them hard to select, design and implement.

Accountability is essential in our target domain. An accountable system provides mechanisms to understand the state of the SUD, and that, in turn, requires a guarantee that all significant actions and events are recorded and that it is possible to evaluate and trace both the state of the system and the reasons why it reached that state.

Auditability is a key requirement for the gambling domain, but also for many IoT applications and for other domains. It goes beyond accountability because it not only guarantees that we record traces of actions and events, but also that the SUD can provide trustworthy proofs of the authenticity of those traces for (normally external) entities called auditors. In this case, it is frequent to have dedicated interfaces and mechanisms for

auditors to access the proofs while maintaining the privacy and confidentiality of the system data.

Finally, transparency refers to a more general scenario, in which the facts, logs and/or reports of the SUD can be provided to any actor requesting them [10]. Obviously, transparency is usually limited or at least affected by privacy and confidentiality.

Moreover, in addition to technical requirements, our target scenarios are also subject to laws and regulations. In the case of the gambling domain, we find very different laws, such as the Gambling Act 2005 [11] for the UK, the Spanish Gambling Act or Decree-Law 422/89 or the Nevada Revised Statutes (mainly the NRS 463 series, such as the NRS 463.0129 "Public policy of state concerning gaming; license or approval revocable privilege").

Regulations governing IoT applications continue to evolve around the world, as governments gain a better understanding of the field and as they develop and evolve laws to protect citizens' rights in the expanding IoT landscape. It is not the goal of this section to provide a comprehensive overview but to highlight that regulations are converging and that the monitoring of relevant events, their protection and their use are part of those regulations. An extensive review of the laws and regulations affecting the IoT around the world is provided in [12]. As a relevant summary, we will discuss the cases of the EU and the USA.

In the European Union, the Cybersecurity Act [13] proposes a voluntary cybersecurity certification framework, which allows companies to certify their ICT products, services and processes according to EU standards. On the other hand, opposed to the voluntary character of the Cybersecurity Act, the recent EU Cyber Resilience Act [14] imposes legal obligations for stringent cybersecurity requirements on digital products, including IoT devices, to be fulfilled from design to marketing.

The European NIS2 Directive [15] extends the scope of the original NIS Directive, to include more sectors and types of entities that must comply with stricter cybersecurity standards. This directive had to be translated into national-level laws by member states by 17 October 2024. The NIS2 Directive proposes more concrete measures to reduce discrepancies between countries and ensure a common high level of cybersecurity across all member states. In particular, it establishes the obligation to facilitate "regular and targeted security audits carried out by an independent body or a competent authority" and further elaborates by establishing that "the results of any targeted security audit shall be made available to the competent authority". Finally, it establishes that "costs of such targeted security audit carried out by an independent body shall be paid by the audited entity", which means that it is in the interest of the entities managing those IT or IoT systems to implement measures to facilitate the audits and reduce the costs associated with them.

In the USA, on 12 May 2021, a Presidential Executive Order on Improving the Nation's Cybersecurity commisioned the definition of two labeling programs on the cybersecurity capabilities of Internet of Things (IoT) consumer devices and software development practices from the National Institute of Standards and Technology (NIST), which, in February 2022, released a White Paper on "Recommended Criteria for Cybersecurity Labeling in the IoT" [16] that contains guidance for IoT manufacturers and integrators. The content of this guide is voluntary, but it is important to mention that, apart from the basics, such as unique device identifiers (a serial number, for example), data protection or firmware updates, it refers to the need to maintain a cybersecurity event log. In particular, it mandates that "the IoT product captures and records information about the state of IoT components that can be used to detect cybersecurity incidents affecting or affected by IoT product components and the data they store and transmit".

2.2. Background

In this section, we provide background information that may be helpful to better understanding the SARMOM pattern. Readers who are familiar with these technologies can safely skip this section.

2.2.1. Message-Oriented Middleware (MOM)

The term Message-Oriented Middleware (MOM) is used to describe a software and/or hardware infrastructure used to send and receive messages between distributed systems and to provide storage (persistence) for those messages. The MOM decouples application components, so that they can use different operating systems and network protocols [17,18]. An important feature of modern MOM implementations, such as Apache Kafka, is that they provide persistent storage and mechanisms to organize messages, guarantee message queue properties (such as immutability, authenticity or ordering), balance the load between MOM servers, etc.

Figure 1 below shows a simplified architecture of an MOM. In particular, the figure does not show the distributed aspects of the MOM, but the logical structure. It is normal that an MOM is distributed among several servers (usually called "brokers"). The figure shows that messages are organized into "topics" and "partitions". A topic is a logical way to separate messages for processing, and each topic is maintained in at least one partition (sometimes more). According to the Kafka documentation "*Each partition is an ordered, immutable sequence of records that is continually appended to—a structured commit log. The records in the partitions are each assigned a sequential id number called the offset that uniquely identifies each record within the partition* ". To improve scalability and throughput, partitions may be managed in different replicas.

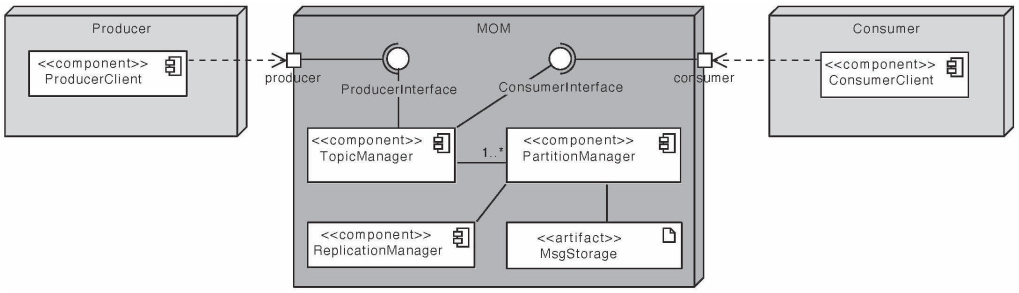


Figure 1. Architecture of a Message-Oriented Middleware (MOM).

Message-sourcing (i.e., Event-sourcing) is an effective way to maintain the state of an application, while at the same time greatly increasing trust in the current state of the system and opening the system to new and enhanced functionalities. Although they share some similarities, an MOM used in this way is not the same as blockchain. If used correctly, this solution can be made tamper-resistant, just like blockchain. Also, like blockchain, data can only be appended. But those are the main similarities. The main difference with blockchain is that this solution can scale to several millions of Transactions Per Second (TPS), while the fastest blockchains reach a maximum of 65,000 TPS.

2.2.2. Secure Registries

From a high-level perspective, and in the field of IT, a registry is a (software/hardware) component, which is frequently used as a tool for system traceability and accountability. Registries provide interfaces to add information records, to consult the registry contents and provide storage capabilities, together with the mechanisms to guarantee the immutability and ordering (sometimes partial) of the records. The information format and size is greatly dependent on the environment where the registry is deployed, ranging from short text log records to any other type of textual or binary information. The access rate to the registry content also varies from one context to another. Built-in security measures within the registry should allow information consumers to verify and trust the information source and content, whereas it also helps to meet the overall system security and regulatory compliance requirements. Thanks to all these features, a secure registry becomes a valuable (if not the most valuable) asset for system (external) audits, which, usually, is the main purpose of a secure registry; hence, it is also called auditable.

Although regular security measures can be applied in order to provide a secure registry, there currently is an ongoing trend regarding the use of blockchain technology to achieve the desired security for registries. Blockchain itself may not be the only security measure, but it is commonly found as one layer in securing the information against unauthorized changes, the key feature of blockchain [19]. In this way, records added to the registry are kept within blocks, integrating hash values to ensure the chain (and, hence, the information it contains) order. This mechanism, together with the decentralization inherent to blockchain, makes it impossible to alter the stored records in any way, given the computational power of current computers. As a result, a secure registry intended for audits is provided.

2.3. Pattern Description Formats

Patterns in software engineering are documented using various formats, to ensure clarity and consistency. The most common formats include the following:

- Alexandrian format: Inspired by Christopher Alexander's A Pattern Language, it includes context, forces, solution and resulting context. This format highlights the balancing of competing forces in design [20].
- Gang of Four (GoF) format: Introduced by Gamma et al. in Design Patterns: Elements of Reusable Object-Oriented Software, this format includes sections like Name, Intent, Applicability, Structure, Participants, Consequences and Implementation [21].
- Pattern-Oriented Software Architecture (POSA) format: Introduced by Buschmann et al. [22], this format emphasizes the pattern's name, context, problem, solution and resulting context, with examples and consequences.
- Security patterns format [23]: This format, specific to security patterns, extends the typical pattern format with sections like Problem, Solution, Security Mechanism, Known Uses, Consequences and Potential Attacks. It focuses on how security concerns are addressed within a pattern.

Each format is tailored to fit the specific domain or design concern, ensuring clear communication of the pattern's intent, applicability and implications. In this paper, we use a variant of the security patterns format for the description of the (SARMOM) pattern and the Alexandrian pattern format for other related patterns.

2.4. Patterns for IoT

While many general patterns are applicable in IoT environments (for instance, encryption and key management patterns) new patterns specific to the IoT field are emerging to meet the specific challenges that arise due to its complex, distributed and heterogeneous nature. Design patterns have been proposed for the IoT, to address issues like scalability, interoperability, security and energy efficiency. The following survey papers summarize the advancements in design patterns for the IoT. Muccini et al. [24] presented a comprehensive collection of architectural patterns, focusing on scalability, energy efficiency and security. Washizaki et al. [25] included an extensive collection of 143 IoT architecture and design patterns for the IoT. On the specific aspect of security, Rajmohan et al. [26] reviewed common security patterns used to mitigate various IoT security threats.

3. The Pattern Ecosystem

SARMOM is a component of a larger ecosystem that is under development. However, there are some patterns that are already available and valuable for this context.

Figure 2 shows our Legacy-to-IoT secure ecosystem and includes some related patterns for generic secure IoT systems developed by other authors. For this diagram, we used UML and represent patterns as stereotyped collaborations. As the definition of collaboration in UML is "a society of classes, interfaces, and other elements that work together to provide some cooperative behavior that's bigger than the sum of all its parts", we believe that it is a perfect fit for representing a pattern using UML, and, therefore, we propose this representation to be adopted by the pattern community. Moreover, a collaboration is

further specified using class and object diagrams, to show their structural aspects, and using sequence and activity diagrams, to show their dynamic aspects. This is coherent with the way patterns have been described in the literature [8].

We note that the proposed UML-based representation does not alter the essence of the pattern diagrams used so far, but simply allows that representation to be integrated in UML, making it possible to use standard tools to create and manage such diagrams and to include other modeling aspects not directly visible in diagrams, thus becoming an integral part of the UML system documentation and enabling the possibility of automated processing of those diagrams.

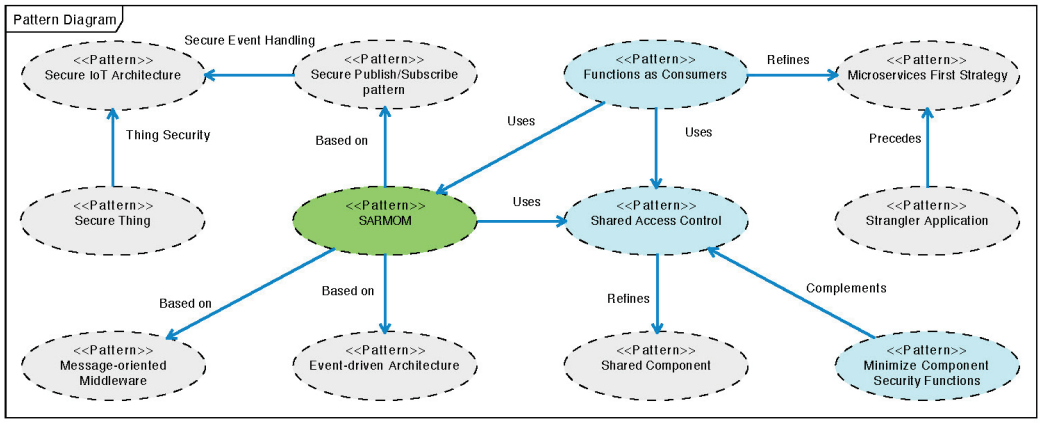


Figure 2. Pattern diagram relating the new pattern with other IoT-relevant patterns.

The patterns in blue in Figure 2 represent patterns that are part of our pattern language, with the SARMOM pattern highlighted in green. The rest of the patterns in gray are related patterns from other authors. The SARMOM pattern is based on Message-oriented Middleware [27], Event-Driven Architecture [6] and Secure Distributed Publish/Subscribe pattern for IoT [28], which, in turn, is related to Secure IoT Architecture and Secure IoT Thing [29]. SARMOM differs from Secure Distributed Publish/Subscribe pattern for IoT in several aspects. In particular, the former follows a strictly pull-based approach, while the latter can use a push-based strategy for messages, which was not convenient in our case. Additionally, SARMOM is tailored for auditability and includes mechanisms to ensure the immutability and ordering of the message chain. Moreover, Secure Distributed Publish/Subscribe pattern for IoT involves the use of Secure Channel, Authenticator, Signer, Encryptor, Decryptor and Signature Verifier patterns, which are not necessary in SARMOM. In particular, the authentication and access control functions are externalized in SARMOM. We rely on the Shared Access Control pattern, which, in turn, is a specialization of Shared Component and complements Minimize Component Security Functions, by outsourcing the access control functionalities, thus minimizing the security functions that are incorporated in SARMOM.

For the sake of completion, the pattern landscape (and the corresponding diagram) includes several patterns that we adopted in conjunction with the Secure Auditable Registry in the process of re-engineering the SUD to become an IoT SUD. These patterns are relevant in the process of implementing the SARMOM pattern. We first adopted Implement Monolith First [6] (not shown in the diagram because we do not consider it necessary in all our target systems), in order to provide maximum value for the clients as an MVP (Minimum Viable Product). We then proceeded to adopt the Strangler Application pattern [6] by means of Proxy Monolith Components to Microservices [6], and then defined a variant for MOM-based SUDs that we call Functions as Consumers, which is meant to be used when implementing SUDs that adopt SARMOM.

In order to facilitate the understanding of some of the patterns that we are still developing, and which are, therefore, not yet available in the literature, we provide brief descrip-

tions for the Shared Access Control, Minimize Component Security Functions and Functions as Consumers patterns. Other solutions that could become patterns in the pattern language are still being revised, and it is not yet decided whether they will be developed as patterns.

3.1. Shared Access Control

An SUD involving different hardware devices is transitioning from monolith to IoT using a microservice architecture. Access to the new services must be controlled, but we need to make sure we provide consistent access control policies. At the same time, we want to avoid data duplication and to reduce the management overhead that would result from using independent access control mechanisms in each service.

Therefore,

we factor the access control functionalities to a common set of services and components that serve the remaining services.

Related patterns:

Shared Component [30], Sidecar [8], XACML Architecture [31], Role-Based Access Control (RBAC) [28], Access Token [32].

3.2. Minimize Component Security Functions

Each component or service in a distributed system has different security requirements, which means different security functions. We want to minimize system redundancy, avoid inconsistent treatment of security, reduce security management overhead and reduce attack vectors. The size of a component is strongly correlated with its security risks. Reducing the component size is good for understandability (which, in turn, improves security). Increasing security by means of partitioning code is a proven and mature strategy [33,34].

Therefore,

we identify security functions that can be externalized to a shared component and we move them there. Additionally, we use "by-design" approaches to reduce the number of security functions required by a service. Finally, we reduce the attack surface of each service by dividing functionalities into separate components, to avoid a service having to implement several security-critical functions.

Related patterns:

Minimize The Functionality of A Component [35], Minimization Principle [36], Shared Component [30].

3.3. Functions as Consumers

We are in the process of transitioning our SUD from a monolith to a microservice event-driven architecture. We want to avoid situations in which new functions continue to be added to the monolith because, on the one hand, we want to avoid duplication of work (those functions will eventually have to be extracted to microservices in the future), and, on the other hand, when implemented in the monolith these functions will become affected by the level of coupling of the monolith and will, therefore, introduce changes in other functions of the monolith that may have already been implemented as microservices, and their understanding and subsequent extraction will become more difficult and risky.

Therefore,

we require that all new functions are specified as consumers of the MOM (i.e., specifying the function as a consumer that retrieves its input data from the MOM) to facilitate and encourage developers to implement them as loosely coupled services fed by information from the MOM.

Related patterns:

Microservices First [6], Replace as Microservice [6].

4. The Secure Auditable Registry Based On Message-Oriented Middleware (SARMOM) Pattern

This section presents the SARMOM pattern using a variant of the security pattern format used in [8]. The reader is referred to [8], in order to obtain a description of each of the pattern sections.

4.1. Intent

Build your system around a Secure Auditable Registry if your system must maintain a record of its operations and must also prove compliance to an external auditor. Complement it by using a Message-Oriented Middleware as the system hub connecting all components, so that all events/messages are consistently treated and can be accessed by the registry.

4.2. Context

The different components of a distributed mission-critical IoT application need to exchange large amounts of information, but we want to maintain these components loosely coupled. The application information can be represented as messages or events. Most information is produced by a specific type of component (i.e., *Things*) and then consumed by multiple components that provide different functionalities. The application operates in a regulated environment and needs to provide access to external auditors, to prove compliance with the applicable regulations.

4.3. Problem

An IoT application is used to obtain event information from different things or machines (e.g., devices and sensors). The amount of information generated is enormous and the rate of production is also extremely fast. The generated information is diverse and is used for many different purposes. Part of the information must be protected, to preserve privacy, to protect business intelligence or because it is considered confidential information, but at the same time it must also serve as proof of the integrity of the system and its information for external auditors.

To represent the problem and the target SUDs, we investigated other systems and domains that could potentially benefit from adopting this pattern. One of those is part of a collaboration we have with an autonomous drone company. Autonomous drones are used for many different tasks, but because they are not controlled by humans their actions must be auditable in every moment. The gambling system and the drone system have very little in common at the functional level, and while there are similarities at the level of components and architecture we found that the main reason for the use of the pattern is not related to the components or the architecture but to the goals and restrictions of the system. For this reason, in order to represent the target SUDs that could benefit from using SARMOM, we concluded that we had to complement the normal UML (class, components or architecture) diagrams showing the architecture of the target systems by further specifying the SUD goals. This was achieved by means of the Secure Tropos Goal Analysis model [37]. Secure Tropos is an independent formalism that is not part of the standard UML. Consequently, to be able to integrate the Secure Tropos model into our modeling environment we developed a specific UML profile to represent Secure Tropos in UML. This profile has been made available as an Open Source artifact.

Figure 3 shows a simplified and generalized architecture of target SUDs before the application of the SARMOM pattern. The depicted architecture is a generalization of several systems that are candidates for the application of the SARMOM pattern. In particular, we have considered scenarios for gambling, autonomous drones and a trusted

mesh communication network for military vehicles. Please note that the dashed lines represent dependencies and do not imply any direction in the communication.

In all these systems, there are "Machines" (i.e., gaming machines, drones or military vehicles) that generate the events that are central to the system operation. Each machine has a controller component that is used to integrate it with the SUD. Machines may send events to a central component (Event Collector) that has a central event store or to specific modules controlling different functions of the SUD, which may also have local event stores. An SUD Management module also exists, to coordinate the rest of the SUD.

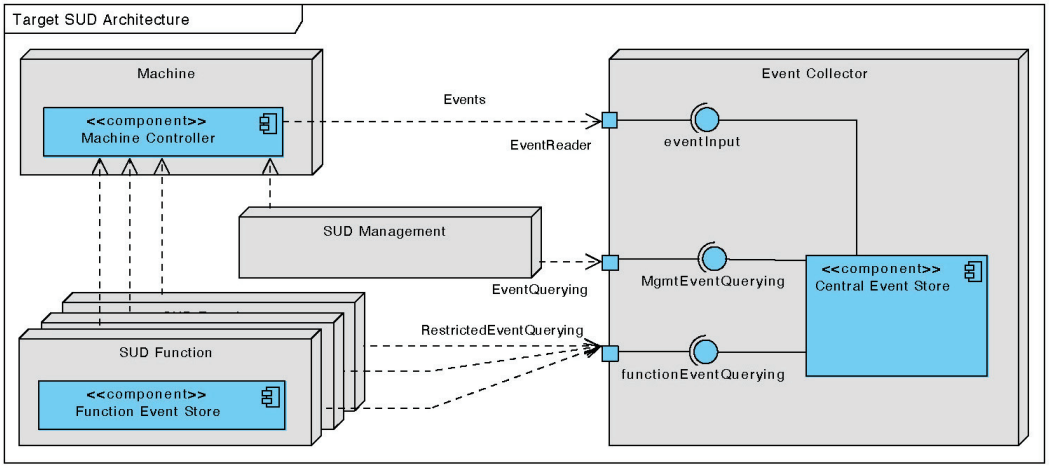


Figure 3. Architecture of SUDs that may use the SARMOM pattern.

While the previous figure may allow the readers to determine whether their SUD can benefit from the application of the SARMOM pattern, we feel it misses important information about the goals of the system if it is taken as the only source for identifying target SUDs for the SARMOM pattern.

To complement the previous architecture diagram, in Figure 4 we present the Secure Tropos Goal Analysis diagram highlighting goals of legitimate and malicious agents and showing how the proposed pattern can help mitigate most of the threats (represented with the «Threatens» relation stereotype) and contribute positively to legitimate goals for the SUD. The diagram uses a representation based on our own UML Profile (the ITIS-UMA-SwEngProfile), which extends UML with different security aspects: for instance, allowing us to represent Secure Tropos concepts and diagrams in UML. In this profile, the stereotype «Agent» is used to represent legitimate stakeholders of the SUD, while «Malicious Agent» denotes entities with illegitimate interests in the system. «Goal» and «Malicious Goal» are used as high-level objectives of agents and malicious agents, respectively. «Mitigation Measure» is a secondary objective, included in the SUD to counteract a «Malicious goal». Relations «+» and «−» indicate, respectively, a positive or negative contribution of one goal to another goal. The «Threatens» relation is used to indicate that one malicious goal is focused on preventing another legitimate goal, and the «mitigates» relation is used to indicate that one legitimate goal is focused on preventing another illegitimate goal. Finally, the stereotypes «include», and «extend» have the standard UML meaning.

The OT/IoT SUD agent represents the legitimate stakeholders of the SUD in general, who need to manage machines in their SUD, and who are also interested in goals such as obtaining SUD intelligence to maximize SUD efficiency and to optimize operations. Guaranteeing operations and the resilience of the SUD are also important goals that may be negatively affected by the marketing goal of lowering the cost of the system. Privacy is a goal of the SUD stakeholders but also of the end users. Another goal of the end users is to gain trust in the SUD, which is positively influenced by the goal of external auditors (representing authorities and regulatory boards) to audit the SUD. It is important to note that this legitimate goal of auditing the SUD can represent a threat to privacy. Likewise, it is important to note that some agents may act selfishly or dishonestly.

For instance, malicious end users may want to circumvent accountability mechanisms to avoid being held accountable for their actions.

The diagram shows how using SARMOM can contribute to mitigating most of the threats and can also contribute positively to achieving goals such as guaranteeing operations and proving correctness.

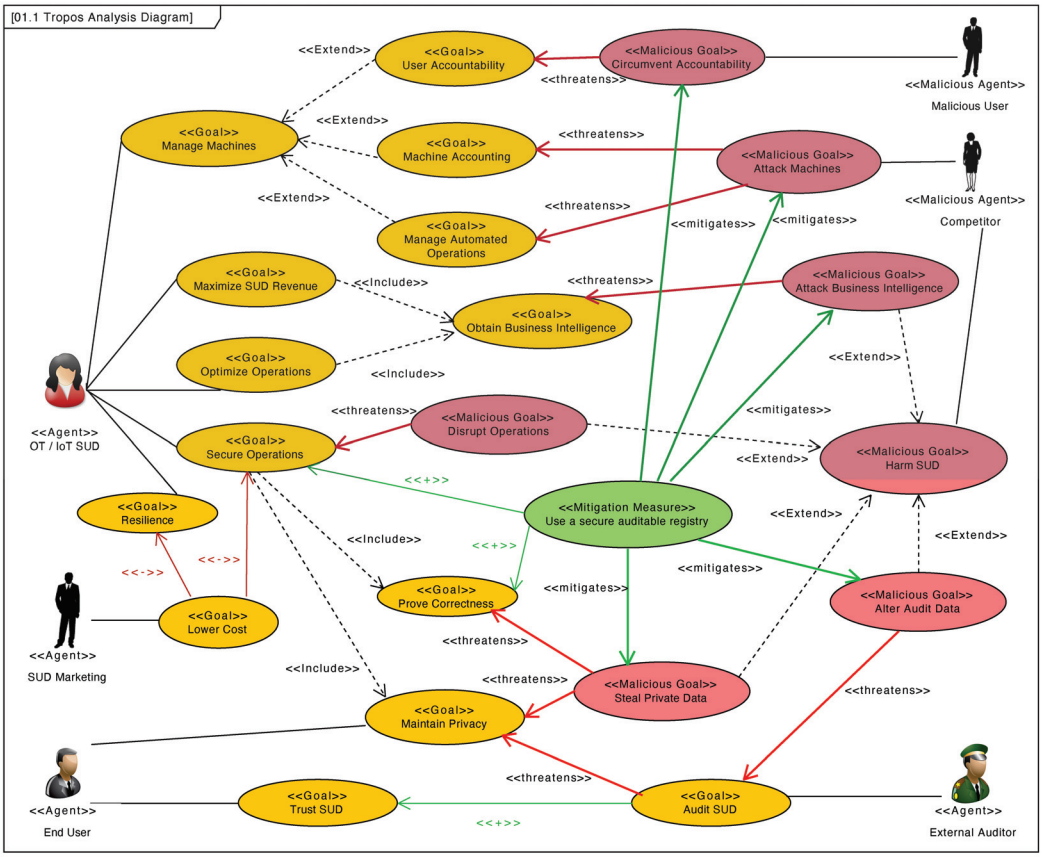


Figure 4. Secure Tropos diagram describing the goals of a typical SUD that may use the SARMOM pattern.

4.4. Solution

The solution consists in using a Secure Auditable Registry (SAR) service that is supported by Message-Oriented Middleware (MOM). The architecture of the system after applying the solution is depicted in Figure 5 below. Please note that the Machine Controller node will be replicated.

A Machine Controller node can manage one machine (i.e., an IoT device), but can also manage a group of machines. In our specific case study, we use a Machine Controller node for a group of slot machines called a "block" that can consist of up to 128 machines, but the pattern can be used with any arbitrary number of machines per controller. In fact, for other domains, such as autonomous drones, there is one Machine Controller per machine (drone).

SUD Client nodes represent the SUD microservices that need to access the data in the registry to provide their functions. Examples of these nodes are Reporting, Accounting, Business Analytics and Anomaly Detection. It is possible to have more than one Auditing Authority node.

The solution entails several steps that can be taken incrementally:

1. In the first place, a Secure Auditable Registry (SAR) service should be integrated with the SUD, using an independent component (e.g., microservice). This component implementing SAR must also implement security mechanisms to protect the integrity and the ordering of the records it contains. See implementation section for details.

2. Then, to free the SAR component from dealing with other requirements, such as concurrent access, scalability and resilience, the access to the SAR must be conveyed through an MOM service. In our case, the ideal solution is to use a pull-based MOM (such as Kafka) so that the registry can retrieve the messages at its own pace, knowing that there is no risk of messages being lost because of the lower performance of the SAR component. In this way, we externalize the management of topics and we have guarantees for the ordering of messages (at least, at the topic level). Additionally, dealing with issues like concurrency, high input throughput and differences in production and consumption speeds is also solved by the MOM. The MOM also helps with error recovery, since it can provide persistence of the records received.

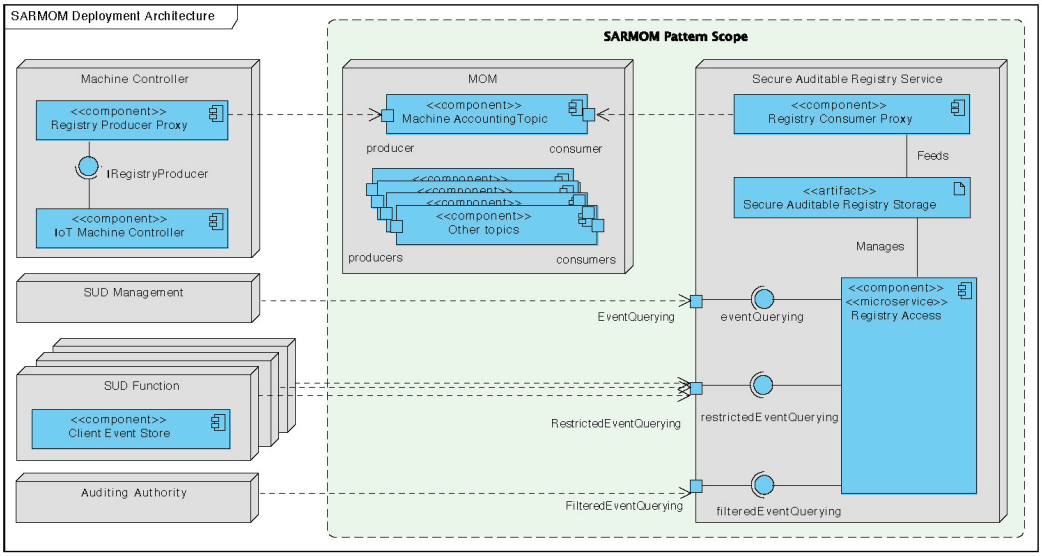


Figure 5. System architecture after applying the SARMOM pattern.

4.5. Implementation

As stated above, a proper context for the SARMOM pattern requires a Message-Oriented Middleware to decouple the secure registry from the target IT system. Even though there are several middleware solutions capable of event streaming, we will refer to Kafka for describing the most relevant implementation considerations that the pattern adopter must take into account. The conclusions in this section are also valid for other MOM implementations.

One key aspect is complexity of configuration and management. Since Kafka supports a distributed operation but only handles the actual data streams and connectivity to clients, a configuration manager component is required. Prior to version 2.8, the Kafka cluster required Zookeeper to manage its configuration. In April 2021, Kafka became self-sufficient, simplifying the architecture by consolidating all configuration management into Kafka itself.

Kafka also provides a set of key–value pairs in a properties file for configuration, which can be supplied either from a file or programmatically. There is a high number of supported properties, which are out of the scope of this work, allowing to configure any aspect or behavior of the middleware: brokers, topics, producers, consumers, connectivity, streams, etc.

The flexibility and versatility of Kafka makes it an excellent choice for many domains, and, consequently, there is a huge ecosystem around it. These properties make Kafka complex to manage (see Refs. [38,39] for details).

The security of the SAR component is also very dependent on the way this component is implemented. The main aspects to consider are the security of the hash chaining system and the security of the registry storage. The ideal implementation for this is to apply a

chained-hash approach [40], in which each record includes the secure hash of the previous record, similarly to what is done with blockchain technologies but without the need to have an infrastructure of peers, conflict resolution, distributed verification, etc. As already mentioned, additional security mechanisms (encryption, anonymization, pseudonymization, redundancy, etc.) can be adopted, depending on the specific requirements of the record contents. The service must also provide secure storage for the records.

The interfaces of the SAR component must be designed not to provide any modification or deletion functions. Only append and read interfaces must be provided. Error handling in this setting must be done by appending error-detection and eventually error-correction events. For instance, a slot machine may report wrong values for its meters (in the gambling domain, meters record things like the amount of money cashed in and out, the number of games played, etc.). When this is detected, the system is not allowed to change the records stored in the registry to reflect the correct values. Instead, new error-detection and error-correction records are inserted, to allow client services to reconstruct the correct state of the system. In this way, the authenticity of the events and the accuracy of the state of the system are guaranteed and can be proved.

Two main read interfaces must, at least, be included. One is a filtered querying interface (FilteredEventQuerying) specifically tailored for the auditing authorities to access (read) the registry contents that are relevant for their role while respecting the privacy requirements. The other one is an unfiltered querying interface, allowing unrestricted read access to the registry contents for the management of the system (named SUD Management in the figure). Please note that this is a conceptual definition of the interface, but, normally, the unfiltered querying interface will be implemented by means of smaller interfaces, which may be restricted for some management functions that do not require full access, following the Interface Segregation Principle. In Figure 5, we have illustrated this idea by including two interfaces, an unrestricted interface (EventQuerying) for the SUD Management, and a restricted interface (RestrictedEventQuerying) that supports only the necessary access for the different SUD Functions. This interface can be further divided when necessary.

Having different SUD Functions with different capabilities may require setting up access control mechanisms, at least for these interfaces, ideally using an external access control service (hence the suggestion to use a shared access control pattern). This access control component is not shown in the diagram because it is not strictly necessary for SARMOM, because its implementation is very dependent on different criteria and because it is covered by external patterns. The materialization of the Secure Auditable Registry (SAR) component as a service allows the target SUD to reduce coupling and facilitates the development of new functions, as they all will use the registry as the main source of data.

4.6. Known Uses

Currently, there are many systems in a variety of industries using Message-Oriented Middleware, such as Kafka, RabbitMQ and Amazon SQS. Many of these systems must also support auditing processes, which means they also need to use different forms of secure registries, a combination that fits the SARMOM pattern proposed in this work.

Below, we list some known uses of systems that combine the use of a Message-Oriented Middleware and a Secure Registry [40]:

- Netflix deals with different IoT devices (ranging from set-top boxes to smart TVs) and uses an MOM system (to convey its real-time monitoring and to support its event-processing pipeline) and a secure registry that is audited by independent processes implemented using HIVE [41].
- Uber deals with car-tracking devices and end-users' devices, and it uses MOM as part of the overall infrastructure stack powering various online and near-real-time use cases. Moreover, Uber uses the distributed cluster of Kafka nodes as a registry when performing internal audits and investigations [42]. While the security needs that arise from the requirement to prove the event sequence to external entities are not

all present there, the internal auditors still need the most important ones that we have identified for the SAR component, referenced in the implementation section.
- Several banks (e.g., Barclays and Rabobank) connect to devices such as ATMs and use both an MOM and a registry that provides the event recording and the audit functions of our SAR to support several financial processes and services, including those for proving compliance and for fraud tracking. Their combined use also fits the present pattern.

4.7. Consequences

After applying the pattern, we can expect different consequences, both positive and negative. Among the positive consequences, we can highlight the fact that we can solve our original problem with additional qualities, such as scalability, performance, ease of evolution and management of a large number of concurrent producers. More specifically, the SAR is the core to achieving the requirements of secure system registry and auditability, while the MOM is the key to achieving the applicability of the concept of the SAR to our target scenarios. Other positive consequences of the application of the SARMOM pattern include the following:

- Enabling asynchronous messaging from producers to the registry through the MOM allows the system to accept high rates of input messages regardless of the consumption capability of the registry. It also improves integrity and fault tolerance.
- MOM is normally an externally developed system, which frequently has control over the policies for evolution of features and security patching.
- The use of the SAR as the source of information for other SUD functions makes the system less vulnerable to data-alteration attacks.
- Likewise, centralizing the sourcing of information in the registry facilitates the adoption of centralized and consistent access control for the different SUD functions (clients).
- Centralizing the sourcing of system events in the registry facilitates the adoption of AI-powered monitors for early detection and prevention of attacks.
- The architecture of the pattern fosters the decoupling of components, thus facilitating maintenance and evolution.
- The intermediation of the MOM makes the registry independent from the source data formats. In particular, most MOM implementations provide the means to add specific interfaces (connectors).

On the downside, we can highlight the following considerations:

- It becomes necessary to install, configure and manage a relatively complex infrastructure, such as the MOM.
- The fact that the MOM is normally an externally developed system reduces the control over the policies for evolution of features and security patching and creates a dependency that may affect our system in case the MOM is discontinued.
- Despite the security mechanisms adopted, the SAR becomes a single point of failure and a single point of attack.
- If not carefully designed, the centralization of the sourcing of information through the SAR may have a negative impact on the performance.

It is important to note that most of the negative considerations can be reduced or eliminated through careful design and integration strategies.

4.8. Threats

Apart from the specific threats described in the previous sections, we present a brief threat analysis in the domain of IoT systems, and, in particular, on their messaging systems. Security is always a major concern regarding distributed applications, and this is especially true for IoT systems. Due to their openness and having their components loosely coupled, IoT systems are continuously exposed to vulnerabilities and threats in different layers and scopes. For messaging-based IoT systems, threats cover aspects such as interceptions, mod-

ifications or even interruption of the services, but also time altering, evidence elimination, etc. The most common threat categories for these systems are the following:

- Unauthorized access: the first fear regarding the use of an IoT system (regardless of whether it is centered on a registry or not) is the unauthorized access to the stored data, allowing attackers to read confidential information, possibly also involving data tampering, corruption or destruction attacks. These attacks have evolved from simple data corruption or injection to become more sophisticated and difficult to detect, and for a system relying on a messaging or registering system the consequences can be fatal.
- Denial of Service (DoS): a frequent goal for attackers is to prevent clients from accessing the resources or services of the system. The most common way to achieve this is by flooding the network or the application servers with a huge amount of information and traffic. For instance, it may result in the registry being unable to write new information or access the information already stored.
- Integrity in multi-user access scenarios: most IoT systems operate in a multi-user environment—that is, a scenario involving several users accessing the same information and/or resources (typically a database or registry) simultaneously. Users can be humans, devices or other IT systems. Keeping data integrity in this context (not only resulting from malicious actions but also accidents and errors) is a key concern. In registry-based systems, integrity also affects the timing and ordering of events and messages.

4.9. Known attacks

The threats above are not just theoretical risks but have actually materialized as attacks on several IoT systems. A cyber attack targeting a critical infrastructure can be catastrophic or even fatal. This section mentions a few recent attacks that were related to the SARMOM pattern, to help the reader understand the pattern better [43,44].

- The Triton attack (2017) was an attack on industrial cyber security systems, in particular, on a Schneider's Triconex safety instrument that was injected with malicious code that could eventually led to the release of toxic gas, which would have been fatal to human life. SARMOM would have contributed to preventing this attack by means of the minimization of data exchange between components of the SUD.
- Data leakage from a North American casino (2017). In this case, attackers gained access to a casino system via a high-tech "smart" fish tank featuring internet connectivity for remote management (temperature, salinity and automated feedings). Although quickly detected, this attack succeeded at retrieving up to 10 gigabytes of data, including the casino's database of high rollers. SARMOM would have contributed to preventing this attack by means of the isolation and centralization of data exchange between components of the SUD.
- The Oldsmar cyberattack on the US water system (2021) targeted the water treatment system of Oldsmar, Florida, gaining access to the control process of the treatment system. The attackers were successful in increasing the sodium hydroxide content in the water systems to toxic levels. However, a human operator was alert and immediately reduced the toxin levels in the water system. SARMOM would have prevented this attack by means of the trusted event sourcing and by minimization of data exchange between the components of the SUD. Moreover, monitoring capabilities enabled by SARMOM would have made it possible to detect the problem and stop the attackers before they could do harm.
- In the Colonial Pipeline cyberattack (2021), the control system of a pipeline carrying jet fuel and gasoline in the US was attacked by ransomware, and the infrastructure was down for nearly six days before service restarted. Although this attack focused on the billing infrastructure of the company, it brought down the entire oil pipeline as they could not issue customers with appropriate bills, affecting the entire east coast of the US. SARMOM would have made this attack much less severe, by means of the

independence of the SUD components, which would have allowed it to contain the effects of the attack in the affected subsystem and would have allowed the system to continue in operation, even if producing bills was not possible, because consumption data would have remained safely stored in the registry for later processing.

5. Conclusions and Future Work

In this paper, we have presented a pattern for a Secure Auditable Registry based on Message-Oriented Middleware to be used in large-scale IoT systems that require audit from external entities. This pattern is part of a pattern language that we are developing for transitioning from legacy systems dealing with hardware devices to IoT. These patters have been applied in an industry-funded R&D project focused on the re-engineering of a legacy high-security system in the domain of gambling. We have worked on generalizing and describing the pattern, so that it is usable in other emerging IoT-enabled domains, such as autonomous vehicles, military operations, banking, healthcare, smart cities, transportation, etc.

Regarding our future work, we plan to work on extending the pattern ecosystem by completing the development of the patterns referred to in Section 1.2 and adding new patterns as they are identified in our current IoT projects. We also plan to prepare computer-oriented versions of these patterns based on COSSPs [45] and to use those to develop pattern-oriented extensions for system engineering tools, to facilitate the adoption of those patterns.

Author Contributions: Conceptualization, A.M.; Software, L.G.; Validation, F.J.J.; Investigation, A.M., F.J.J. and L.G.; Writing—original draft, A.M.; Writing—review & editing, F.J.J. and L.G.; Project administration, A.M.; Funding acquisition, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by the Junta de Andalucía, Spain, under contract QUAL21 010UMA.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Straits Research. Internet of Things (IoT) Market: Information by Component (Hardware, Connectivity, Services), End-User Vertical (Manufacturing, Transportation, Healthcare), and Region—Forecast Till 2030. 2022. Available online: https://straitsresearch.com/report/internet-of-things-market (accessed on 19 June 2024).
2. Al-Sarawi, S.; Anbar, M.; Abdullah, R.; Hawari, A.B.A. Internet of Things Market Analysis Forecasts, 2020–2030. In Proceedings of the 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), London, UK, 27–28 July 2020; pp. 449–453. [CrossRef]
3. Vailshery, L.S. *Number of Internet of Things (IoT) Connected Devices Worldwide from 2019 to 2021, with Forecasts from 2022 to 2030*; Statista: Hamburg, Germany, 2022. Available online: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/ (accessed on 10 September 2024).
4. Priyadarshini, U. Real World IoT Applications in Different Domains. 2022. Available online: https://www.edureka.co/blog/iot-applications/ (accessed on 19 June 2024).
5. Cartwright, I.; Horn, R.; Lewis, J. Patterns of Legacy Displacement. 2022. Available online: https://martinfowler.com/articles/patterns-legacy-displacement/ (accessed on 19 June 2024).
6. Brown, K.; Woolf, B.; Groot, C.D.; Yoder, J. Cloud Adoption Patterns. 2021. Available online: https://kgb1001001.github.io/cloudadoptionpatterns/ (accessed on 19 June 2024).
7. Fernandez, E.B.; Washizaki, H.; Yoshioka, N.; Okubo, T. The design of secure IoT applications using patterns: State of the art and directions for research. *Internet Things* **2021**, *15*, 100408. [CrossRef]
8. Fernandez, E.B. *Security Patterns in Practice: Designing Secure Architectures Using Software Patterns*; Wiley Publishing: Hoboken, NJ, USA, 2013.
9. *ISO/IEC 25010:2011*; Systems and Software Engineering—Systems and Software Quality Requirements and Evaluation (SQuaRE)—System and Software Quality Model. ISO: Geneva, Switzerland, 2017. Available online: https://www.iso.org/standard/35733.html (accessed on 19 June 2024).

10. Anisetti, M.; Ardagna, C.; Damiani, E.; Maña-Gomez, A.; Spanoudakis, G. Towards transparent and trustworthy cloud. *IEEE Cloud Comput.* **2017**, *4*, 40–48. [CrossRef]
11. Gambling Act 2005. Latest Available Revised Version. 2005. Available online: https://www.legislation.gov.uk/ukpga/2005/19/contents (accessed on 19 June 2024).
12. Thales. IoT Cybersecurity: Regulating the Internet of Things. Thales Digital Identity and Security. 2024. Available online: https://www.thalesgroup.com/en/markets/digital-identity-and-security/iot/inspired/iot-regulations (accessed on 10 September 2024).
13. European Economic and Social Committee. Opinion of the European Economic and Social Committee on the 'Proposal for a Regulation of the European Parliament and of the Council on ENISA, the "EU Cybersecurity Agency", and Repealing Regulation (EU) No 526/2013, and on Information and Communication Technology Cybersecurity Certification ("Cybersecurity Act")'. Regulation (E.U.) 2019/881 of 17 April 2019 (COM(2017) 477 final/2 2017/0225 (COD)) OJ C, C/227, 28.06.2018. 2018; p. 86. Available online: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.C_.2018.227.01.0086.01.ENG (accessed on 10 September 2024).
14. European Commission; Directorate-General for Communications Networks, Content and Technology. Cyber Resilience Act—New EU Cybersecurity Rules Ensure More Secure Hardware and Software Products 2024. Available online: https://data.europa.eu/doi/10.2759/543836 (accessed on 10 September 2024).
15. European Parliament. Directive (EU) 2022/2555 of the European Parliament and of the Council of 14 December 2022 on Measures for a High Common Level of Cybersecurity Across the Union, Amending Regulation (EU) No 910/2014 and Directive (EU) 2018/1972, and Repealing Directive (EU) 2016/1148 (NIS 2 Directive). L 333/80 EN Official Journal of the European Union 27.12.2022. 2022. Available online: https://eur-lex.europa.eu/eli/dir/2022/2555/oj (accessed on 10 September 2024).
16. National Institute of Standards and Technology. *Recommended Criteria for Cybersecurity Labeling for Consumer Internet of Things (IoT) Products*; Nist Cybersecurity White Paper; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2022. [CrossRef]
17. Liu, Y.; Zhang, L.J.; Xing, C. Review for Message-Oriented Middleware. In *Internet of Things—ICIOT 2020*; ICIOT 2020. Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12405, pp. 12–23. [CrossRef]
18. Fu, G.; Zhang, Y.; Yu, G. A Fair Comparison of Message Queuing Systems. *IEEE Access* **2021**, *9*, 421–432. [CrossRef]
19. Rajasekaran, A.S.; Azees, M.; Al-Turjman, F. A comprehensive survey on blockchain technology. *Sustain. Energy Technol. Assess.* **2022**, *52*, 102039. [CrossRef]
20. Alexander, C. *A Pattern Language*; Oxford University Press: Oxford, UK, 1977.
21. Gamma, E.; Helm, R.; Johnson, R.; Vlissides, J. Design patterns: Abstraction and reuse of object-oriented design. In Proceedings of the ECOOP'93—Object-Oriented Programming: 7th European Conference, Kaiserslautern, Germany, 26–30 July 1993; Proceedings 7; Springer: Berlin/Heidelberg, Germany, 1993; pp. 406–431.
22. Buschmann, F.; Henney, K.; Schimdt, D. *Pattern-Oriented Software Architecture: On Patterns And Pattern Language*; John Wiley & Sons: Hoboken, NJ, USA, 2007; Volume 5.
23. Schumacher, M.; Fernandez-Buglioni, E.; Hybertson, D.; Buschmann, F.; Sommerlad, P. *Security Patterns. Integrating Security and Systems Engineering*; Wiley: Hoboken, NJ, USA, 2006.
24. Muccini, H.; Moghaddam, M.T. IoT architectural styles: A systematic mapping study. In Proceedings of the Software Architecture: 12th European Conference on Software Architecture, ECSA 2018, Madrid, Spain, 24–28 September 2018; Proceedings 12; Springer: Berlin/Heidelberg, Germany, 2018; pp. 68–85.
25. Washizaki, H.; Ogata, S.; Hazeyama, A.; Okubo, T.; Fernandez, E.B.; Yoshioka, N. Landscape of Architecture and Design Patterns for IoT Systems. *IEEE Internet Things J.* **2020**, *7*, 10091–10101. [CrossRef]
26. Rajmohan, T.; Nguyen, P.H.; Ferry, N. A decade of research on patterns and architectures for IoT security. *Cybersecurity* **2022**, *5*, 2. [CrossRef]
27. Fehling, C.; Leymann, F.; Retter, R.; Schupeck, W.; Arbitter, P. *Cloud Computing Patterns*; Springer: Berlin/Heidelberg, Germany, 2014.
28. Fernandez, E.B.; Astudillo, H.; Orellana, C. A pattern for a Secure IoT Thing. In Proceedings of the 26th European Conference on Pattern Languages of Programs (EuroPLoP'21), Graz, Austria, 7–11 July 2021; Association for Computing Machinery: New York, NY, USA, 2022; pp. 1–6. [CrossRef]
29. Fernandez, E.B.; Uzunov, A.V. Secure middleware patterns. In Proceedings of the 4th International Symposium on Cyberspace Safety and Security (CSS 2012), Melbourne, Australia, 12–13 December 2012; pp. 470–482.
30. Steel, C.; Nagappan, R.; Lai, R. *Core Security Patterns: Best Practices and Strategies for J2EE, Web Services, and Identity Management*; Pearson: Hong Kong, China, 2005.
31. Fernandez, E.B.; Yoshioka, N.; Washizaki, H. Secure Distributed Publish/Subscribe (P/S) Pattern for IoT. In Proceedings of the AsianPLoP, Online, 12–16 October 2020.
32. Richardson, C. *Microservices Patterns With Examples in Java*; Manning Publications: New York, NY, USA, 2018.
33. Huang, Z.; Jaeger, T.; Tan, G. Fine-grained program partitioning for security. In Proceedings of the 14th European Workshop on Systems Security, Scotland, UK, 26 April 2021; pp. 21–26.
34. Maña, A.; Pimentel, E. An Efficient Software Protection Scheme. In Proceedings of the IFIP SEC'01, Paris, France, 11–13 June 2001; Kluwer Academic Publishers: Dordrecht, The Netherlands, 2001.
35. Wheeler, D.A. Secure Programming HOWTO. v3.72 Edition. 2015. Available online: https://dwheeler.com/secure-programs/Secure-Programs-HOWTO.pdf (accessed on 19 June 2024).

36. Jackson, C.; Russell, S.; Sons, S. The Information Security Practice Principles. Foundational Whitepaper. Version 0.9. Indiana University Center for Applied Cybersecurity Research. 2017. Available online: https://cacr.iu.edu/principles/ISPP-Foundational-Whitepaper-2017.pdf (accessed on 20 October 2024).
37. Paja, E.; Dalpiaz, F.; Giorgini, P. STS-tool: Security requirements engineering for socio-technical systems. In *Engineering Secure Future Internet Services and Systems: Current Research. Lecture Notes in Computer Science (LNCS)*; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8431, pp. 65–96.
38. Kreps, J. Kafka Ecosystem. 2022. Available online: https://cwiki.apache.org/confluence/display/KAFKA/Ecosystem (accessed on 19 June 2024).
39. Powered by Kafka. Latest Revised Version. Available online: https://kafka.apache.org/powered-by (accessed on 19 June 2024).
40. Han, M.; Jiang, W. A Secure Communication Method Based on Message Hash Chain. *Appl. Sci.* **2022**, *12*, 4505. [CrossRef]
41. Confluent. Featuring Apache Kafka in the Netflix Studio and Finance World. 2020. Available online: https://www.confluent.io/blog/how-kafka-is-used-by-netflix/ (accessed on 19 June 2024).
42. Blog, U.E. Presto® on Apache Kafka® At Uber Scale. 2022. Available online: https://uber.com/en-ES/blog/presto-on-apache-kafka-at-uber-scale/ (accessed on 19 June 2024).
43. Jain, S.; Lakshmi, V.; Srivathsa, D.R. *IoT and OT Security Handbook*; Packt Publishing: Mumbai, India, 2023.
44. Lee, M. Criminals Hacked a Fish Tank to Steal Data from a Casino. Forbes Newsletters. 2017. Available online: https://www.forbes.com/sites/leemathews/2017/07/27/criminals-hacked-a-fish-tank-to-steal-data-from-a-casino/ (accessed on 19 June 2024).
45. Mana, A.; Fernandez, E.B.; Ruiz, J.F.; Rudolph, C. Towards Computer-oriented Security Patterns. In Proceedings of the 20th International Conference on Pattern Languages of Programs PLoP'13, Monticello, IL, USA, 23–26 October 2013.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Article

Realization of Authenticated One-Pass Key Establishment on RISC-V Micro-Controller for IoT Applications

Tuan-Kiet Dang *, Khai-Duy Nguyen, Binh Kieu-Do-Nguyen, Trong-Thuc Hoang and Cong-Kha Pham

Department of Computer and Network Engineering, University of Electro-Communications (UEC), 1-5-1 Chofugaoka, Tokyo 182-8585, Japan; khaiduy@vlsilab.ee.uec.ac.jp (K.-D.N.); binh@vlsilab.ee.uec.ac.jp (B.K.-D.-N); hoangtt@uec.ac.jp (T.-T.H.); phamck@uec.ac.jp (C.-K.P.)
* Correspondence: tuankiet@vlsilab.ee.uec.ac.jp

Abstract: Internet-of-things networks consist of multiple sensor devices spread over a wide area. In order to protect the data from unauthorized access and tampering, it is essential to ensure secure communication between the sensor devices and the central server. This security measure aims to guarantee authenticity, confidentiality, and data integrity. Unlike traditional computing systems, sensor node devices are often limited regarding memory and computing power. Lightweight communication protocols, such as LoRaWAN, were introduced to overcome these limitations. However, despite the lightweight feature, the protocol is vulnerable to different types of attacks. This proposal presents a highly secure key establishment protocol that combines two cryptography schemes: Elliptic Curve Qu–Vanstone and signcryption key encapsulation. The protocol provides a method to establish a secure channel that inherits the security properties of the two schemes. Also, it allows for fast rekeying with only one exchange message, significantly reducing the handshake complexity in low-bandwidth communication. In addition, the selected schemes complement each other and share the same mathematical operations in elliptic curve cryptography. Moreover, with the rise of a community-friendly platform like RISC-V, we implemented the protocol on a RISC-V system to evaluate its overheads regarding the cycle count and execution time.

Keywords: IoT; secure communication; key establishment; signcryption; ECQV; RISC-V

1. Introduction

In modern society, the utilization of smart devices and services has become more prevalent. These encompass a vast range of industries, including home automation, manufacturing, medicine, finance, and transportation. The Internet of things (IoT) refers to the billions of interconnected devices that are equipped with sensors and actuators that can be accessed from virtually any location in the world through the Internet. A wireless sensor network (WSN), which is a subset of an IoT network, is a network that incorporates a large number of wireless sensor nodes, wireless communication, and data processing functions to monitor, gather, and transmit information from the environment. In contrast to the Internet, the construction of wireless sensor networks relies on low-complexity communication protocols. Low-power and wide-area communication networks, such as Zigbee and LoRa, are preferable for the development of WSNs, as they satisfy the constrained conditions of small sensor nodes. As the wireless network continues to expand, these nodes are scattered over a large area and are prone not only to physical attacks but also to cyberattacks. Therefore, it is crucial to provide security measures for WSNs just as much as the security measures for the Internet.

A typical wireless sensor node is equipped with a sensing device, a wireless transceiver, a power management unit, and a central computing unit. The node devices in a WSN are inherently resource constrained, especially in batteryless and energy-harvesting applications: they have limited processing speed, storage capacity, and communication bandwidth. In a

typical network, the sensor nodes communicate between themselves using radio signals. The constraints limit the integration of the high-security standard protocols used among modern computers, like SSL (secure sockets layer) or TLS (transport layer security). As mentioned, long range (LoRa), which is a widely adopted network in low-power applications, is a radio modulation technology that is capable of offering long-range, low-power, and secure data transmission for IoT devices. The LoRaWAN MAC protocol specifies the regulations for LoRa and defines the standards for LoRa networking. However, despite the lightweight feature, the LoraWAN protocol still faces significant threats from malicious parties [1–3]. Han and Wang [4] indicated that the flaws of LoRaWAN lie in the key management and data confidentiality. Lacking a mechanism for session key establishment enables long-term leakage of the root key due to side-channel attacks in encryption and decryption with the AES cipher block. Such potential weaknesses of LoRaWAN outweigh the lightweight feature of applying the symmetric AES-128 cipher. Several attempts were proposed to improve LoRaWAN's security measures. Naoui et al. [5] proposed applying the concept of the session key to update the secret key every session of communication, thus enhancing the secure communication channels between the end node and the network server. Han and Wang [4] proposed applying a lightweight Rabbit cipher in the key derivation function to the root key update scheme, and hence, strengthening the security of session keys.

The lightweight feature of symmetric key cryptography is an essential property that a key establishment (KE) protocol for WSNs should possess. Other proposals also tried to benefit from symmetric key cryptography and add more security properties to establish secure communication channels. For example, Pu et al. [6] proposed a lightweight KE protocol that provides an authentication property. The protocol applied a Tinkerbell chaotic map, a physical unclonable function (PUF), and a one-way hash function to provide an authentication session key agreement. Similarly, Zheng and Chang [7] proposed a mutual authentication protocol that also relies on a hash and a PUF primitive for the key exchange between IoT endpoints. These methods and the LoRaWAN share a common principle that exchanging messages in the public domain requires a cryptographic hash function with optimal collision resistance properties. In other words, the auxiliary materials for the key establishment are hashed and then sent to other parties through the handshake. The confidentiality of the materials is guaranteed by the hash function. Researchers have also tried to achieve a lightweight feature for secure communication of IoT devices with stronger cryptography.

Highly secure cryptography, such as public key cryptography (PKC), has been known to be expensive due to the limited computing resources of sensor nodes. The traditional Diffie–Hellman key exchange used in modern computers requires an authentication feature to prevent man-in-the-middle attacks. Torres et al. [8] indicated that IoT devices cannot afford the significant computational cost. Additionally, node devices need to update their public and private keys to prevent ciphertext-only attacks similar to the root key revelation of LoRaWAN. On the other hand, Keoh et al. [9] suggested that a careful selection of settings for public-key-based ciphersuites would ensure the security of the IoT network and reduce overheads in the handshake. Sciancalepore et al. [10] claimed that public key cryptography has become feasible on the latest generation of constrained devices and supported this statement by proposing an elliptic curve cryptography (ECC)-based protocol for IoT devices. Moreover, other than LoRaWAN, recent works for highly sensitive data transmission (healthcare, e-payment) applied PKC with various IoT communication technologies, like ZigBee, RFID, and NFC. A. Rehman et al. [11] utilized the Zigbee wireless technology to establish a reliable network management (RNM-SC) for a smart healthcare system. A group of sensor nodes are managed by a local coordinator and connected to a ZigBee router. The system offers an optimization in the transmission path, which is handled by the router, and a node table comprising information on close-distance nodes. The underlying key establishment to ensure authenticity relies on the station-to-station algorithm, which is based on the classic Diffie–Hellman method. Furthermore, RNM-SC applies Blum–Goldwasser asymmetric cryptography to encrypt and decrypt sensitive

data. An extensive review by I. E. Gaabouri [12] addressed the security challenges and threats of systems based on RFID and NFC technologies. It is suggested that the deployment of a strong cryptography scheme is a feasible solution to reinforce RFID's communication security. The review also summarized the most recent studies on cryptography schemes for RFID and NFC and showed a diversity of encryption methodologies for solutions implemented for different applications. Based on the statistical data, the author suggested that cryptography systems like ECC are preferable for semi-active and active tag devices. Meanwhile, a lightweight scheme is favorable for passive tag devices where resource limitations are more strict. In addition, many recent works shared the same methodology of integrating strong cryptography, like ECC, to secure communication in IoT networks [13–15].

Our proposal applies two schemes that can be constructed by using ECC. When constructing a protocol that uses ECC-based handshakes for node devices with small processors and low-bandwidth transceivers, there are several important factors to consider. First, a key establishment protocol that is proposed under a specific operating assumption must ensure security properties, such as confidentiality and authentication. Second, the number of computations executed at the node side is a critical issue that should be addressed. Third, the size and number of exchange messages should be reduced for low-bandwidth communication. With limited power and energy on a batteryless or self-charging system on an IoT node, the second and the third factors should be carefully addressed to improve the power allocation between the processing unit and the transceivers, as well as lengthen the operation of the whole system under limited power conditions.

Based on these speculations, this paper proposes three major contributions:

- This research was the first to propose utilizing a signcryption key encapsulation mechanism in composition with an Elliptic Curve Qu–Vanstone (ECQV) implicit certificate scheme [16] to construct a two-level key establishment protocol. By applying a hybrid cryptosystem, the proposed protocol preserves the security characteristics of both schemes and takes advantage of both public key cryptography and symmetric cryptography. Depending on the demand of the security level, the protocol, at the very least, can enable a fast rekeying method of session keys derived from a generated key pair, allowing for sending session key generation materials and encrypted sensor data in one message exchange. At a higher level of security, the protocol allows for updating the public and private key pair; hence, only several sessions share a key pair to derive the session keys. At the highest security level, every communication session can issue a fresh key pair and session key to communicate.
- Under a two-party communication model, our approach integrates a physical unclonable function to the ECQV scheme to provide additional authentication data to the handshake. This lightweight solution enhances the ECQV scheme and allows for only authenticated node devices to request a key pair. Thus, the modified ECQV scheme can protect against common attacks, such as impersonation attacks and replay attacks.
- We implemented a 32-bit RISC-V system that is suitable for a resource-constrained sensor node to assess the computational overheads of the proposed protocol. The software version of the protocol was tested on different configurations of the RISC-V processor to evaluate the timing cost. Moreover, instead of using software-based or pseudo-security primitives, the RISC-V system is equipped with standard physical security primitives, including a true random number generator (TRNG) and a PUF.

The remainder of this paper is organized as follows: Section 2 presents the background of the paradigm of a hybrid cryptosystem and the selected schemes. It also summarizes the mathematical notions used in the proposed protocol. Section 3 introduces the assumption of the system model for an IoT network and the combination of the ECQV implicit certificate scheme and the one-pass signcryption key exchange mechanism. This section also describes step-by-step operations in the proposed protocol. Section 4 analyzes the protocol's security features at different levels of security. Section 5 compares the communication and computation cost of the proposed protocol with related works that also support

authentication property. Section 6 discusses the protocol's advantages and limitations and outlines the directions for future work. Finally, section 7 concludes the paper.

2. Preliminary Background

This section briefly reviews the ideas of a hybrid cryptosystem, ECQV implicit certificate scheme, and signcryption key establishment mechanism. In addition, a summary of the mathematical notions used in the proposed protocol is provided.

2.1. Hybrid Cryptosystem

A key establishment protocol aims to allow two participants who have their own source of randomness and need to establish a common secret. The output shared secret must be the same value for the two participants, and it has to be unknown to any outsider. The proposed key establishment protocol is designed based on a hybrid cryptosystem, which was formalized by Cramer and Shoup [17]. The hybrid construction paradigm, also known as the KEM-DEM construction, consists of two lower-level building blocks: a key encapsulation mechanism (KEM) and a data encapsulation mechanism (DEM). Generally, a KEM is a method by which two participants exchange their random contributions to the generation of the shared secret in a protected manner. Specifically, a KEM takes a private and public key pair to generate a random symmetric key and its encapsulation for the handshake. Meanwhile, a DEM properly encrypts a message payload using the generated symmetric key (the shared secret) with a symmetric block cipher. Fundamentally, a KEM consists of three operations:

- A key generation operation *KEM.KeyGen* that outputs a public and private key pair. The generation of a key pair depends on the chosen underlying algorithm.
- An encapsulation operation *KEM.Encrypt* that takes a key pair of the sender, and a public key of the receiver, and outputs a shared key K and an encapsulated text $C1$.
- A decapsulation operation *KEM.Decrypt* that takes a key pair of the receiver, a public key of the sender, and an encapsulated text $C1$, and extracts the shared key K if the operation is valid.

A data encapsulation mechanism (DEM) is either an encryption or a decryption operation of a chosen block cipher. Figure 1 describes the model of a KEM and DEM in a hybrid cryptosystem.

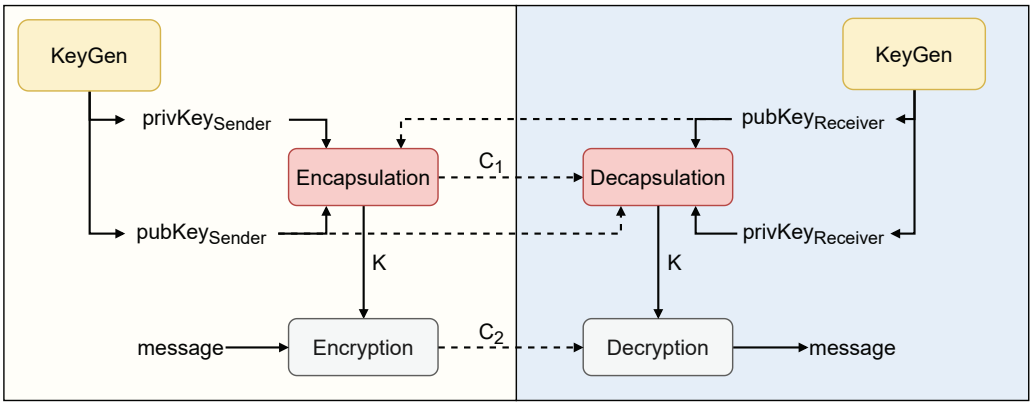


Figure 1. Hybrid KEM-DEM paradigm.

This hybrid paradigm takes advantage of the speed and efficiency of symmetric cryptography in the encryption and decryption of an arbitrary length payload. Moreover, the hybrid model relies on highly secure public key cryptography to generate a unique symmetric key for communication. This modular design approach also has a primary benefit in that the security requirements of the asymmetric and symmetric parts of the scheme can be completely independent of each other. Thus, it allows for freedom in the

selection of the underlying algorithms for each part to construct a protocol that suits a specific application [18,19]. An example of a cryptosystem based on the Diffie–Hellman (DH) method that applies an AES cipher block as the data encapsulation can be found in [20]'s specification. Note that these selections for a hybrid cryptosystem offer security to an extent under a particular security assumption; thus, it is still vulnerable to some attacks, such as key compromise impersonation attacks. This research aimed to utilize the modular characteristic of this hybrid model to construct a key establishment protocol. The proposed protocol applies a modified version of a KEM with signcryption cryptography. Also, the ECQV implicit signcryption scheme is used as the *KEM.KeyGen* operation to reduce the cost of public key cryptography as opposed to a DH-based KEM in the specification [20].

2.2. ECQV Implicit Certificate Scheme

As mentioned above, the KEM requires a public–private key pair before the encapsulation operation. In public key cryptography, the Diffie–Hellman key exchange is the common method for two parties to generate their key pairs and exchange the public parts. This key exchange requires explicit signatures from both participants to authenticate the exchange materials. However, generating these certificates can be computationally intensive, which is not ideal for IoT node devices. The ECQV scheme leverages the mathematical properties of ECC to associate IoT nodes' identities with their public keys, forming implicit certificates. This characteristic reduces the computational effort on authentication from one party in the communication channels in contrast to traditional public key cryptography methods, such as RSA, ECDSA, or EdDSA, which require both parties to compute and directly verify each other's certificates. In the context of a WSN, the ECQV scheme provides a passive approach, ensuring that only the deployed sensor nodes can derive the correct key pair from the implicit certificate issued by the trusted server. This scheme effectively mitigates the computational overhead associated with the key pair generation and authentication on the node devices. Moreover, the scheme reduces the quantity and size of the exchange messages in the network.

2.3. Signcryption

Signcryption is a public key authenticated encryption introduced by Zheng [21]. The key idea behind signcryption is to integrate encryption and signature operations into a single computation, thereby reducing the computational overhead and improving the efficiency compared with performing the operations separately. The cryptography primitive aims to simultaneously achieve confidentiality, integrity, and authentication at a lower cost. Dent [22–24] extended the principles of signcryption to the key encapsulation mechanism and provided security definitions for the integration. The resulting signcryption KEM (SKEM) became a variant of the signcryption primitive capable of generating authenticated symmetric keys. The characteristics of SKEM motivated the exploration of the connection between SKEM and one-pass key establishment (OPKE) by Gorantla et al. [25]. Based on the one-pass HQMV scheme [26], Gorantla et al. proposed SKEM-based OPKE, which provides a highly efficient method to generate ephemeral symmetric keys with confidentiality and authenticity. This study aimed to realize the practicality of OPKE based on SKEM to meet the demands of wireless sensor networks.

The proposed protocol applies elliptic curve cryptography as the mathematical foundation of both the ECQV and SKEM schemes. Therefore, the parameters of a selected curve can be shared for the two schemes. For a specific curve, let $\mathcal{G}$ be an elliptic curve group, and let $G \in \mathcal{G}$ be a generator of order (prime) q of the group $\mathcal{G}$. The ECQV requires a hash modulo function $\mathcal{H}_1$, which computes the hash and then performs modulo to the order q. The function hash without modulo is denoted as $\mathcal{H}_2$. In addition, the ECQV algorithm is added with a function of a PUF with a challenge–response pair (CRP) denoted as $(\mathcal{C}^t, \mathcal{R}^t)$. Table 1 shows the notions used in the two schemes.

Table 1. Symbols and corresponding meanings.

	Symbols	Corresponding Meaning
Parameters and variables	ID	The identity
	G	The base point of ECC
	q	A prime order of a specific elliptic curve
	$(\mathcal{C}^t, \mathcal{R}^t)$	A challenge $\mathcal{C}^t$ and its corresponding response $\mathcal{R}^t$ at time t
	t, t_n, t_s	Generated random values in $[1, q-1]$
	R, R_n, R_s	Curve points generated from random values
	$Cert_N$	Implicit certificate designated for a node with ID_N
Keys	(k_S, K_S)	A private–public key pair of the server
	(k_N, K_N)	A private–public key pair of a node with ID_N
	K	A symmetric session key
Functions	$\mathcal{H}_1()$	Hash modulo function
	$\mathcal{H}_2()$	Hash function
	$F_{PUF}()$	Function of a PUF
	$Enc()$	Encryption algorithm using the symmetric key K
	$Dec()$	Decryption algorithm using the symmetric key K

3. Proposed Key Exchange Protocol

3.1. Security Model Assumption

The proposed protocol was designed for a wireless sensor network model consisting of an array of IoT sensor nodes attempting to communicate with a trusted server. Each sensor node is assigned an identity ID. The server is also assigned with an ID_S. Some IoT applications, such as medical wearable sensor devices, are required to provide an anonymity property (hiding the true ID of a node device) to the protocol. However, the anonymity property was out of the scope of this study. The server is a legitimate entity responsible for gathering sensor data and distributing materials for the generation of public–private key pairs. In the trusted manufacture domain, the server generates its asymmetric key pair (k_S, K_S). The server's private key k_S must be securely stored on the server side. The server's public key K_S and its identity ID_S are distributed to all sensor nodes in the network before deployment. The proposed protocol implements a physical unclonable function entity in each sensor node. Therefore, the server is required to store the challenge–response pairs $(\mathcal{C}^t, \mathcal{R}^t)$ of all sensor nodes in its database.

3.2. Practical Application of ECQV and One-Pass SKEM Methodologies

Wireless sensor networks are subject to constraints, particularly regarding power and energy consumption, which can range from hundreds of milliwatts to the microwatt scale. We aimed to construct a protocol that supports low-power WSNs while providing sufficient security properties. To achieve this goal, the optimal approach is to reduce the computation on the node devices by applying symmetric block ciphers, such as AES, ASCON, and TEA, which offer a lightweight encryption and decryption process. Furthermore, the limitation in computation was also considered in the key generation process, and SKEM is a suitable scheme to generate and transport the key generation material at the least cost. By definition, the SKEM scheme requires asymmetric key pairs to generate symmetric session keys. Instead of applying the classic Diffie–Hellman method to generate and exchange generation materials for key pairs, the ECQV scheme significantly reduces the computing cost of authentication. As a result, a combination of ECQV and SKEM provided an optimal solution to reduce the computing effort at the node side while still applying strong asymmetric cryptography to establish secure communication.

The idea of the proposed protocol was to utilize the advantages that SKEM offers:

- The session key generation with signcryption gives assurance to the sender that the session key is available only to the recipient and assurance to the recipient that the key came from the sender.

- One-pass key establishment provides a very efficient construction of session keys, which reduces the complexity of the handshake and computation at the sensor nodes.
- The encrypted sensor data can be transmitted simultaneously with key generation materials, which favors the intermittent operation of the sensor nodes.

According to Okamoto [27], it is important for one-pass key establishment protocols to have the security goal of key compromise impersonation. This is also true for OPKE based on SKEM, where only the sender contributes ephemeral data to the symmetric key generation. An adversary who can expose the key pair of the sender can permanently impersonate the sender. Therefore, the practical application of the SKEM-based OPKE relies completely on the secrecy of the asymmetric key pair. Moreover, in the deployment domain, the SKEM scheme publicly exposes the sender's identity ID in the exchange message. Without the existence of a trusted third party (a public key infrastructure) to generate key pairs for the sender and the receiver, any adversary can claim to possess that same ID and establish a shared key with the recipient. In order to lower the complexity of a WSN, the server and the sensor nodes need a method to update the key pairs at the node to mitigate the risk of impersonation.

The ECQV implicit certificate scheme is a low-cost solution to this problem. The authenticity of a key pair yielded by the ECQV scheme adds another layer of assurance to the server that the generated session key is bound to not only the identity of a sensor node and the server but also the key pair designated for that sensor node. The sensor nodes can benefit from the low cost in the size of the exchange message and relieve themselves from the burden of signing and verifying explicit signatures. However, the deployment of an ECQV scheme faces the same issue due to the revelation of the sensor nodes' identities ID_N. In the deployment domain, the ECQV scheme can be triggered by a malicious party claiming to have ID_N. The implicit certificate is bound to the identity, not the physical device that has been assigned with ID_N. From the server's perspective, it can only recognize the identity that is trying to make a request for a certificate stored in the database. Therefore, the proposed protocol resolves this problem by binding the implicit certificate with the identity ID_N and a unique characteristic of an actual physical node device. A physical unclonable function implemented inside of the processor of a sensor node can provide distinctive responses that characterize a sensor node from each other. Furthermore, it is worth noticing that the ECQV scheme allows the server to compute a sensor node's public key without an additional exchange message from the sensor node. The modified ECQV with a PUF makes the public key somewhat "private" when using the response in the computation. In addition, from the standpoint of a sensor node, a matching challenge–response pair guarantees that it is communicating with the authentic server.

The realization of SKEM-based OPKE is beneficial for a constrained WSN. The proposed protocol provides the intermittent operation of batteryless or self-charging IoT nodes with higher secure communication channels. By applying the ECQV scheme as the key pair generation method, and a PUF as an enhancement to the ECQV scheme, a WSN can benefit from the advantages of SKEM-based OPKE with minimal risks.

3.3. Protocol Proposal

This study's primary contribution was the construction of a two-level security protocol when combining ECQV and SKEM schemes to establish secure communication channels in an IoT network. Figure 2 describes the proposed protocol. The two-level security notion implies the dependency of the generation of a session key on only a SKEM or both SKEM and ECQV scheme.

A level 1 secure communication channel, whose security property depends solely on the SKEM scheme, is established when a generated node's key pair (k_N, K_N) is used repetitively to create fresh session keys K. This mechanism enables quick session key generation at the cost of only one exchange message. A feasible application of SKEM requires assumptions on secure storage for ephemeral data and the key pair on the node device to avoid long-term compromises. To eliminate the risk of exposing or compromising

the random source and the key pair stored on a node device, a level 2 secure channel is created with a unique and fresh session key using a newly generated node's key pair. At this level, the handshake to generate a session key requires three exchange messages to sequentially execute both ECQV and one-pass SKEM schemes. An alternative use case is that periodically updating the key pair of a node can minimize the risk of temporary leakage of a key pair when using only the SKEM scheme and benefit from the low exchange message count.

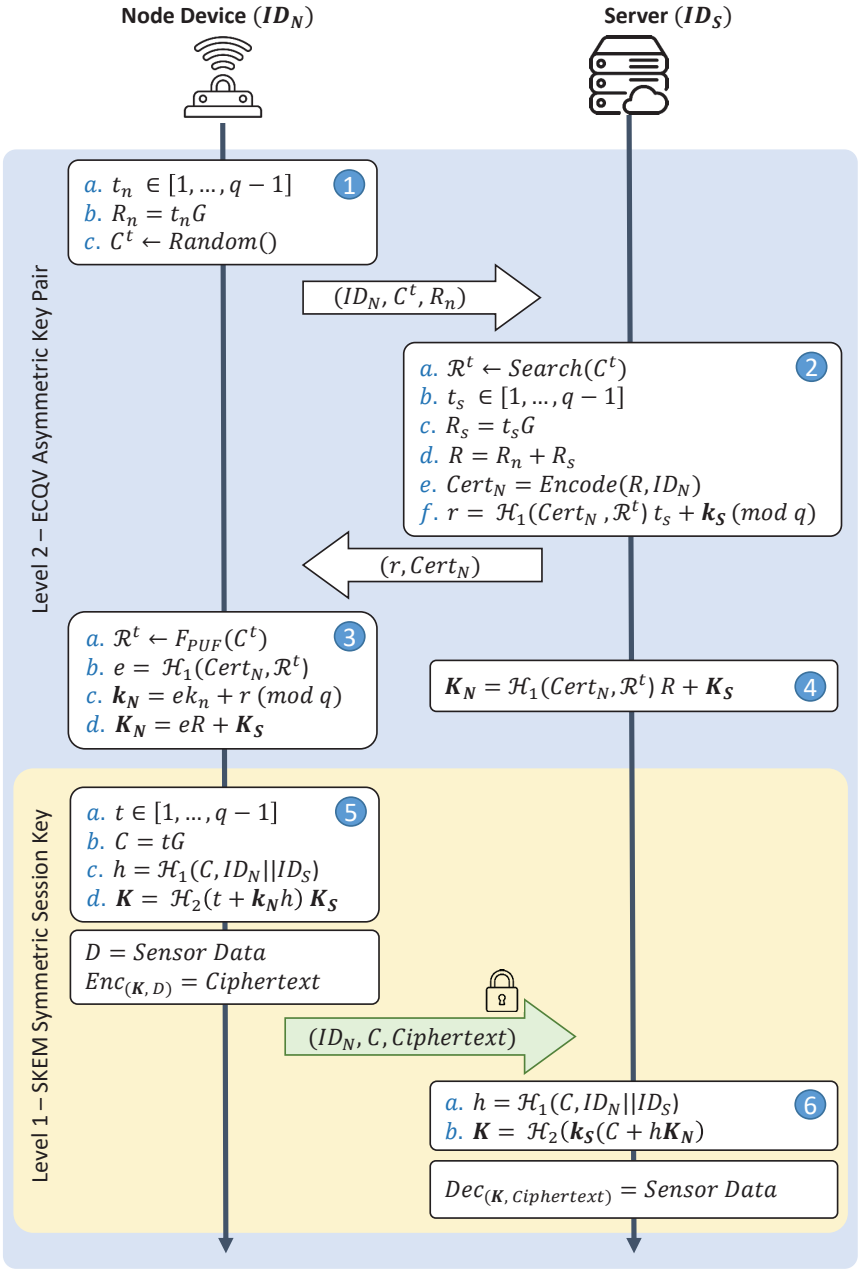


Figure 2. 2-level security key establishment protocol. The computations are grouped into six steps, each indicating sequential operations at the sender and receiver.

3.3.1. Level 2: ECQV Asymmetric Key

The establishment of a secure communication channel starts with the ECQV handshake modified with the PUF function. An implicit certificate is issued by a node device as follows:

- Step 1—Certificate request: A node generates a random number $k_n \in \{0, q\}$ and calculates the corresponding elliptic curve point using $R_n = k_n G$. The random

number k_n must be securely stored at the node side; then, the point R_n, the node's identity ID_N, and a randomly chosen challenge C^t are sent to the server as a certificate request.

- Step 2—Certificate generation: After receiving the request, the server generates another random number and its corresponding point (k_s, R_s). Then, the implicit certificate is generated through hashing an elliptic curve point $R = R_n + R_s$. Upon storing the challenge–response pair, the server computes the implicit signature r with the corresponding response $\mathcal{R}^t$ of the received challenge $\mathcal{C}^t$.

$$r = \mathcal{H}_1(Cert_N, \mathcal{R}^t)k_s + k_S(mod\, q) \tag{1}$$

- Step 3—Key extraction: After receiving the implicit certificate, the node generates the response $\mathcal{R}^t$ from the previously selected challenge $\mathcal{C}^t$. Afterward, it extracts its designated key pair (k_N, K_N) using $\mathcal{R}^t$.

$$e = \mathcal{H}_1(Cert_N, \mathcal{R}^t) \tag{2}$$

$$k_N = ek_n + r(mod\, q) \tag{3}$$

- Step 4—The server can compute the public key of the node with Equations (2) and (3) by itself, thus reducing the cost of a message from the node to share its public key.

As mentioned earlier, depending on the security requirement of IoT applications, the ECQV handshake will be executed to provide the level 2 security channel. Otherwise, the node only executes the SKEM handshake to exchange data with the lowest cost.

3.3.2. Level 1: One-Pass SKEM Symmetric Session Key

In order to establish a level 1 secure channel, the node and the server execute the SKEM scheme in two main steps:

- Step 5—Key encapsulation: The node device generates a random point C on the elliptic curve. Then, the encapsulation of the random point C and the IDs of the node and the server are computed:

$$h = \mathcal{H}_1(C, (ID_N||ID_S)) \tag{4}$$

The session key is created at the node side using the encapsulation, the node's private key, and the server's public key. The session key is computed as follows:

$$K = \mathcal{H}_2((t + k_N h)K_S) \tag{5}$$

With the session key, the node can encrypt its data and send it to the server. Thanks to the one-way design of SKEM, the node can combine the ciphertext and the materials to generate the same session key into a compound message. Then, it transmits the message to the server.

- Step 6—Key decapsulation: After receiving the information, the server retrieves each segment and starts computing the session key:

$$K = \mathcal{H}_2(k_S(C + hK_N)) \tag{6}$$

After the server decrypts the ciphertext, successful communication is guaranteed. A meaningful message indicates that the communication is successful; otherwise, an adversary may have compromised the node. If this is the case, the server can force an ECQV key pair update or wait until the node's key pair expires.

4. Security Analysis

This section presents a security analysis of the proposed protocol. The analysis was conducted by applying the informal (non-mathematical) methodology to provide a direct interpretation of the supported and non-supported security properties of the protocol.

According to the criteria of the secure authentication and key agreement scheme, it satisfies several security properties and can defend against common attacks.

(a) *Mutual authentication*: In the level 1 secure channel, signcryption assures that only the intended participants with valid identities can extract the correct session key K. The modified ECQV scheme guarantees the server that only the sensor node having ID_N and a valid CPR can derive the correct key pair (k_N, K_N) from the implicit certificate. On the other hand, knowing the server is the only one having stored its CRPs, the sensor node can be certain that only the server (with its valid private key k_S) can derive the correct session key K. Therefore, the proposed protocol can achieve mutual authentication even for level 1 security communication channels.

(b) *Integrity*: Both parties using ECQV and SKEM schemes verify their knowledge of a key pair and a symmetric key after successfully using the keys. Thus, the content of the exchange messages can be verified after meaningful communication is conducted. Any modification of the messages will result in invalid keys and meaningless decrypted messages.

(c) *Impersonation attack*:

- *Node impersonation attack*: We assumed that an adversary tries to impersonate a node with ID_N to communicate with the server. At level 1 security, the adversary cannot compute the session key due to lacking a valid key pair (k_N, K_N) that is bound to the ID_N and the PUF's response $\mathcal{R}^t$. While at level 2 security, the adversary cannot replicate a PUF primitive on the node; hence, it cannot derive the valid key pair (k_N, K_N) without knowledge of the response $\mathcal{R}^t$.
- *Server impersonation attack*: With one-pass SKEM, the server does not respond to the sensor node; thus, we assumed that an adversary tries to impersonate the server to issue the implicit certificate. This attack is only possible if the adversary can break into the server and retrieve the CRP database. Otherwise, the certificate issued by the adversary is meaningless and shall result in an invalid key pair (k_N, K_N).

As a result, the malicious adversary cannot impersonate either a legitimate server or an IoT device.

(d) *Known session key attack*: A protocol with this property can prevent an adversary from accessing future communication to the server, even if they obtain one or more previous session keys. In the proposed protocol, the generation of a sensor node's key pair and session keys requires ephemeral random materials. Therefore, if the session keys are revealed, the future communication channels will remain secure.

(e) *Node capture attack*: We assumed that an adversary has captured a node with ID_N. Through physical memory disclosure attacks, the adversary can retrieve stored data on the node device, such as the valid key pair. As a result, the communication with the level 1 channel would be compromised. To eliminate the risk, a level 2 secure channel computes a fresh key pair (requiring new random numbers and shuffling with new CRP) for every communication session. Another approach to mitigate the risk is to periodically update the key pair by triggering the ECQV scheme after a certain period. Consequently, the proposed protocol offers two methods: (a) complete immunity to the node capture attack and (b) a method that minimizes data leaking.

(f) *Perfect/partial forward secrecy*: Perfect forward secrecy is a security property that ensures that compromising the key pair of one or both entities does not lead to the compromise of past session key establishment. Partial forward secrecy refers to a situation where some, but not all, past communications remain confidential, even if the key pairs are compromised. The level 1 secure channel can offer partial forward secrecy in the case of updating a node's key pair after a certain time. In contrast, triggering the ECQV scheme at level 2 security creates ephemeral key pairs, meaning that one session key corresponds to one key pair. A three-pass handshake can ensure perfect forward secrecy.

(g) *Replay attack*: A replay attack is a form of network attack where an attacker intercepts and maliciously retransmits data that was previously recorded. A one-pass SKEM is susceptible to this kind of attack. Gorantla addressed some solutions to this problem by assigning a session ID for every communication session or applying a time stamp [25]. Updating the key pair of a node is a method to mitigate the issue. Therefore, a secure channel at level 2 can completely prevent this kind of attack.

(h) *Message modification attack*: As we discussed regarding the integrity property, if an attacker manages to capture and alter any part of a message being exchanged over a secure channel, it can result in different keys being extracted by both the sensor node and the server. This can also cause both parties to fail to agree upon the same session key. Consequently, the communication is meaningless and will not leak any critical information; therefore, the protocol is immune to modification attacks.

The summary of the security features of our proposal is shown in Table 2, which also includes comparisons with other protocols.

Table 2. Comparison of functionality features of the proposed protocol with related protocols.

Features	Proposed Level 2	Proposed Level 1	[14]	[28]	[29]	[30]	[31]
Mutual authentication	●	●	●	●	●	●	●
Integrity	●	●	●	●	●	●	●
Impersonation attack	●	●	●	●	●	●	○
Session key attack	●	●	●	●	○	●	●
Node capture attack	●	○	○	●	●	●	○
Perfect forward secrecy	●	●	●	●	●	●	●
Replay attack	●	○	●	●	○	●	○
Message modification attack	●	●	●	●	●	●	●
Anonymity	○	○	●	●	○	●	○

● Secure against a particular attack or supports a particular feature. ○ Insecure against a particular attack or does not support a particular feature.

5. Performance Evaluation

This section describes the experimental setup to analyze the performance of the proposed protocol and the existing protocols. The comparison includes the communication and computational costs. The communication costs were evaluated in terms of the number of exchange message and the total size of these messages. The computational costs were evaluated on an embedded system that can be implemented on actual IoT devices. The protocol was tested with a secp256k1 256-bit EC, with a 256-bit size for the SHA2 hash function. The identity of a node ID and a challenge C^t for a PUF primitive were assumed to be 8 bytes.

5.1. Communication Overhead

In a wireless sensor network with battery-powered or self-charging node devices, to seamlessly monitor the environment, the nodes must be consistently active and periodically transmit data [32]. The proposed protocol aimed to assist the monitoring operation by lowering the complexity of establishing secure communication channels. At the minimum cost with level 1 security, the encapsulated material for session key generation was an identity of a node and a random point of the elliptic curve. This message (ID_N, C) cost $8 + 32 = 40$ bytes. The communication overhead counted only the cost to establish the keys; thus, this message size excluded the size of the ciphertext. The computation overhead of a level 2 secure channel was higher compared with level 1, as it required two more messages. The request from a sensor node (ID_N, C^t, R_n) was $8 + 8 + 32 = 48$ bytes. The corresponding implicit certificate $(r, Cert_N)$ was $32 + 40 = 72$ bytes. In total, the handshake at level 2 security cost three exchange messages and 160 bytes. Note that the overall communication cost varied depending on the frequency of reissuing the key pairs.

We compared our protocol with other existing key establishment schemes that also provide authentication properties in Table 3. In [14], the key establishment required two parties to update information for consecutive communication. Excluding the registration phase, the protocol needed four exchange messages, with a total of 168 bytes. The research in [29] proposed two protocols, in which the handshake of the first protocol cost four exchange messages and 420 bytes, while the second one cost two messages and 142 bytes. In [28,30], the size of messages to create an authenticated channel nearly doubled the cost of a proposed protocol at level 2 security with quantity.

Table 3. Communication cost comparison.

Protocol	Number of Messages	Total Message Size (Bytes)
Ours—level 2	3	160
Ours—level 1	1	40
[14]	4	168
[28]	3	316
[29]—Protocol #1	4	420
[29]—Protocol #2	2	142
[30]	3	332

5.2. Computation Overhead

Different approaches can be used to evaluate the computation overhead of a communication protocol. Some studies, like [14,30], tested the cost of elliptic curve operations and hash functions on desktop processors and then deducted the overall timing cost from their proposed protocols. On the other hand, the studies in [28,29] calculated the overall timing cost by adopting the timing of those operations and functions from other works. These methods can only present a relative timing cost without considering the overhead of a system architecture, like bus interfaces and memory transactions. Therefore, we evaluated the proposed protocol by developing a hardware system. The primary purpose of this approach was to precisely assess the workload of executing the protocol on an embedded system suitable for sensor nodes. The open-source RISC-V processor was utilized to measure the computing performance by implementing two micro-architectural configurations. The RISC-V architecture offers a performance counter that can be accessed through the control and status registers (CSRs). Hence, the measurement metric extracted from the CSRs provides information about the actual clock cycle executed by the processor. This data can be used to derive the timing cost based on the operating clock frequency of the system. By analyzing this information, it is possible to gain insight into the costs associated with different architecture designs and identify areas for further architecture optimization.

The current options for processors in IoT devices include ARM, x86, and RISC-V. With the open-source instruction set architecture (ISA), RISC-V ISA provides a high flexibility in customization for a specific application compared with its counterparts. RISC-V enthusiasts have extended the processor with additional instructions and accelerators to enhance the computing efficiency and provide security and low-power design for IoT devices. Rocket is a variant of the RISC-V processor that allows for modifications at the system and micro-architecture levels. Its architecture is suitable for evaluating the performance of the proposed protocol. The performance data can be used to tailor the Rocket core further to realize the practicality of the protocol for resource-constrained IoT nodes.

The software version of the proposed protocol was implemented on a RISC-V system with a Rocket core, which is a variant of the RISC-V core that can be scaled down for low-power sensor nodes. The RISC-V system was synthesized on an Arty A7-100T Xilinx FPGA to evaluate the computation cost regarding a number of clock cycle counts. Instead of using pseudo- or software-based primitives, the RISC-V system integrated the physical hardware implementation of a true random number generator and a physical unclonable function, as shown in Figure 3.

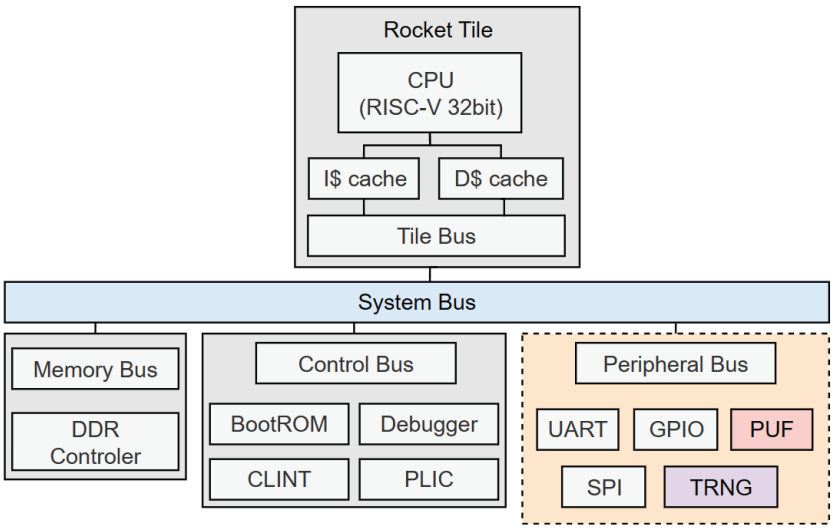


Figure 3. A 32-bit IMAC RISC-V system.

Table 4 shows the notation of computations applied in the proposed protocol and some related works in key establishment based on ECC with an authentication property. Table 5 compares the computation overheads of the protocol at the sensor nodes and the server separately. The proposed protocol aimed to minimize the operations of the sensor nodes as much as possible. The protocol allowed for establishing a level 1 secure channel with the least computational cost $(2T_{sm} + T_a + 2T_h)$. The protocol in [14] yielded the same cost in computing; however, the material to establish a share key was quadrupled. With level 2 security, the computation cost was less than [29,30]. However, the application of the protocol needed to take both the computation and communication of the handshake into consideration.

Table 4. Computation notation and timing of elliptic curve computations.

Notation	Description	RV32-IMAC #Cycle	RV32-I #Cycle
T_h	Hash function	13,943 ± 10	14,178 ± 7
T_{sm}	ECC scalar multiplication	129,821,421 ± 26,894	136,189,184 ± 17,860
T_a	ECC point addition	266,589 ± 32	274,658 ± 13
T_r*	Random number generation	80,649 ± 20	82,349 ± 9
T_{puf}*	PUF function	135,810 ± 14	137,442 ± 28
T_{mod}	Modulo	23,711 ± 4	25,160 ± 18
T_{ba}	Big number addition	698 ± 2	838 ± 1
T_{bm}	Big number multiplication	5009 ± 5	5075 ± 6

* Timing varied depending on the design and system implementation of the primitives.

Table 5. Computation cost comparison.

Protocol	Sensor Node	Sever
Ours—level 2	$4T_{sm} + 2T_a + T_h$	$3T_{sm} + T_a + 3T_h$
Ours—level 1	$2T_{sm} + T_a + 2T_h$	$2T_{sm} + 2T_h$
[14]	$2T_{sm} + 3T_h$	$2T_{sm} + 3T_h$
[28]	$4T_{sm} + 3T_h$	$5T_{sm} + 5T_h$
[29]—Protocol 1	$11T_{sm} + 3T_a + 10T_h$	$4T_{sm} + T_a + 8T_h$
[29]—Protocol 2	$5T_{sm} + 2T_a + 7T_h$	$3T_{sm} + T_a + 7T_h$
[30]	$5T_{sm} + T_a + 16T_h + T_f^*$	$4T_{sm} + T_a + 8T_h$

T_f^*: fuzzy extractor.

Table 4 also presents the measurement results of the EC operations, the hash function, and basic operations on big numbers used in the proposed protocol. The measurement was conducted on two different Rocket cores with two configurations: RV32I and RV32IMAC.

The notations I and IMAC describe the extension of the instruction set architecture used to design the Rocket core. The extension I is the base integer instruction set that can support various operating environments. The extension M contains instructions that support multiplication and division. By default, the RV32I can emulate any extension; however, this experiment aimed to test the significance of other extensions on computing the proposed protocol. The measurement exposed the core operations of each computation to reduce the increase in the cycle count of branching and other operations of the program. It was expected that the ECC scalar multiplication required the most computing effort. The measurement results show that the RV32IMAC with more instruction extensions and hardware cost could achieve a lower cycle count when processing the computations, however the performance improvement was insignificant. In specific, the ECC scalar multiplication could achieve only a 5% increase in performance.

We conducted another timing measurement to assess the execution of the proposed protocol on the Rocket RV32IMAC. The operation of the RISC-V system was expected to be the same as if it was operating at a sensor node. The computations of the protocol were broken down into smaller steps. The timing measurement was conducted to wrap around a function that computed all of the operations in a single step. Table 6 shows the cycle count and execution time of the system running at a 50 MHz clock supply.

Table 6. Timing measurement of individual step on RISC-V 32-bit IMAC.

Proposed Protocol		Computation	#Cycle	Time (ms)
Level 2 ECQV Asymmetric key pair	1.a	$T_r + T_{mod}$	105,810 ± 24	2.11
	1.b	T_{sm}	130,443,888 ± 56,314	2608.88
	1.c	T_r	80,649 ± 20	1.78
	3.a	T_{puf}	135,810 ± 14	2.72
	3.b	$T_h + T_{mod}$	39,103 ± 28	0.78
	3.c	$T_{bm} + T_{ba} + T_{mod}$	30,867 ± 20	0.62
	3.d	$T_{sm} + T_a$	130,789,946 ± 56,350	2615.80
		Subtotal	261,626,073	5232.69
Level 1 SKEM Symmetric session key	5.a	$T_r + T_{mod}$	105,810 ± 20	2.11
	5.b	T_{sm}	130,443,888 ± 56,314	2608.88
	5.c	$T_h + T_{mod}$	39,103 ± 28	0.78
	5.d	$T_{bm} + T_{ba} + T_h + T_{sm}$	130,463,538 ± 56,326	2609.27
		Subtotal	261,052,339	5221.04
		Total	522,678,412	10,453.73

Based on the experimental results, the RISC-V Rocket core with extra extensions and hardware cost could not deliver a noticeable increase in computing efficiency compared with the default ISA. Therefore, the base configuration Rocket RV32I is suitable as the central processing unit for resource-constrained sensor nodes. To further improve the performance of computing the EC operations, an accelerator designed specifically for EC multiplication attached to the Rocket core as a RoCC, an accelerator connected directly to the Rocket core, is a direct and cost-effective solution. The trade-off between the level 2 and level 1 security of the proposed protocol was that the level 2 secure channel offered more security properties at the cost of doubling the computation cost. In contrast, with a level 1 secure channel, a sensor node required less effort to establish a communication channel, thus reducing the active time of the processor and the transceiver. However, the modified ECQV scheme had to be triggered periodically to mitigate the risk of some attacks.

6. Discussion

The proposed protocol offers a hybrid cryptography approach, integrating both asymmetric and symmetric cryptography to leverage the advantages of each. Unlike other solutions that impose high costs in communication and computation in rekeying every session, the protocol provides a lightweight mechanism to reuse the asymmetric key pair

while updating fresh session keys. With these functions, the protocol can still offer essential security properties and prevent common attacks in IoT networks. However, it lacks support for anonymity, which is a critical requirement in highly sensitive data transmission scenarios, such as healthcare, e-payment, and e-voting. Anonymity ensures that the identities of communicating parties remain concealed, thereby protecting privacy and preventing unauthorized tracking. Furthermore, to maximize the benefit of the level 1 security channel, deciding the lifetime of a node's key pair is critical. An optimal update period can maximize the benefit of the fast rekeying mechanism and mitigate the risk of particular attacks like replay attacks. Additionally, despite the intended limitation in computation at the node side, the time required to compute elliptic curve cryptography remains relatively high. Consequently, this limitation restricts the frequency of sensor data transmission periods, particularly in applications requiring the transmission of sensor data every minute. Therefore, we aim to improve the protocol by addressing such limitations in future work at the system's architecture level. Furthermore, a systematic approach to designing a secure system for sensor nodes that encompasses both network and hardware device security is needed. Trusted execution environment (TEE) mechanisms can be integrated to improve a node's security against physical attacks, like side-channel attacks. Safeguarding random sources and protecting the memory storage of keys are mandatory.

7. Conclusions

This paper presents an ECC-based authenticated key establishment protocol for wireless sensor networks. The protocol combines implicit certificate ECQV, one-pass signcryption key encapsulation mechanism schemes, and a physical unclonable function primitive to construct a two-level security protocol. Each level offers a method to establish a secure channel with a trade-off in computation cost, communication cost, and security properties. The security analysis demonstrated that the proposed protocol is able to defend against common attacks on IoT networks. The protocol enables a fast rekeying method with only one message exchange at the minimum cost, which is beneficial for wireless sensor networks. In addition, we evaluated the overhead of the proposed protocol to other existing ECC-based authenticated key establishment protocols. The result shows that our protocol performed better in terms of the computation cost, number, and size of the exchanged messages. The comparison results indicate that the proposed protocol is a competitive approach for achieving highly secure and low-cost communication in wireless sensor networks. Moreover, a timing measurement of the protocol on a 32-bit RISC-V system, which was integrated with two essential hardware primitives (TRNG and PUF), was provided to evaluate the practicality of the protocol on an embedded processor. Based on the measurement results, further improvement on the RISC-V hardware can result in an overall low-cost RISC-V system that supports the protocol for sensing or monitoring applications.

Author Contributions: Conceptualization, C.-K.P.; Methodology, T.-K.D.; Software, T.-K.D.; Investigation, T.-K.D.; Resources, K.-D.N. and B.-K.-D.-N.; Writing—original draft, T.-K.D.; Writing—review & editing, K.-D.N., B.-K.-D.-N., T.-T.H. and C.-K.P.; Supervision, T.-T.H. and C.-K.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Aras, E.; Ramachandran, G.S.; Lawrence, P.; Hughes, D. Exploring the Security Vulnerabilities of LoRa. In Proceedings of the 2017 3rd IEEE International Conference on Cybernetics (CYBCONF), Exeter, UK, 21–23 June 2017; pp. 1–6. [CrossRef]
2. Sundaram, J.P.S.; Du, W.; Zhao, Z. A Survey on LoRa Networking: Research Problems, Current Solutions, and Open Issues. *IEEE Comm. Surv. Tutor.* **2019**, *22*, 371–388. [CrossRef]

3. Loukil, S.; Fourati, L.C.; Nayyar, A.; In, C.S. Investigation on Security Risk of LoRaWAN: Compatibility Scenarios. *IEEE Access* **2022**, *10*, 101825–101843. [CrossRef]
4. Han, J.; Wang, J. An Enhanced Key Management Scheme for LoRaWAN. *Cryptography* **2018**, *2*, 34. [CrossRef]
5. Naoui, S.; Elhdhili, M.E.; Saidane, L.A. Enhancing the Security of the IoT LoraWAN Architecture. In Proceedings of the International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN), Paris, France, 22–25 November 2016; pp. 1–7.
6. Pu, C.; Zerkle, H.; Wall, A.; Lim, S.; Choo, K.-K.R.; Ahmed, I. A Lightweight and Anonymous Authentication and Key Agreement Protocol for Wireless Body Area Networks. *IEEE Internet Things J.* **2022**, *9*, 21136–21146. [CrossRef]
7. Zheng, Y.; Chang, C.-H. Secure Mutual Authentication and Key-Exchange Protocol between PUF-Embedded IoT Endpoints. In Proceedings of the 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Republic of Korea, 22–28 May 2021; pp. 1–5. [CrossRef]
8. Torres, N.; Pinto, P.; Lopes, S.I. Security Vulnerabilities in LPWANs—An Attack Vector Analysis for the IoT Ecosystem. *Appl. Sci.* **2021**, *11*, 3176. [CrossRef]
9. Keoh, S.L.; Kumar, S.S.; Tschofenig, H. Securing the Internet of Things: A Standardization Perspective. *IEEE Internet Things J.* **2014**, *1*, 265–275. [CrossRef]
10. Sciancalepore, S.; Piro, G.; Boggia, G.; Bianchi, G. Public Key Authentication and Key Agreement in IoT Devices with Minimal Airtime Consumption. *IEEE Embed. Syst. Lett.* **2017**, *9*, 1–4. [CrossRef]
11. Rehman, A.; Haseeb, K.; Fati, S.M.; Lloret, J.; Peñalver, L. Reliable Bidirectional Data Transfer Approach for the Internet of Secured Medical Things Using ZigBee Wireless Network. *Appl. Sci.* **2021**, *11*, 9947. [CrossRef]
12. Gaabouri, I.E.; Senhadji, M.; Belkasmi, M.; Bhiri, B.E. A Systematic Literature Review on Authentication and Threat Challenges on RFID Based NFC Applications. *Future Internet* **2023**, *15*, 354. [CrossRef]
13. Mao, G.; Liu, Y.; Dai, W.; Li, G.; Zhang, Z.; Lam, A.H.F.; Cheung, R.C.C. REALISE-IoT: RISC-V-Based Efficient and Lightweight Public-Key System for IoT Applications. *IEEE Internet Things J.* **2024**, *11*, 3044–3055. [CrossRef]
14. Li, B.; Zhang, G.; Lei, S.; Fu, H.; Wang, J. A Lightweight Authentication and Key Agreement Protocol for IoT Based on ECC. In Proceedings of the 2021 International Conference on Advanced Computing and Endogenous Security, Nanjing, China, 21–22 April 2022; pp. 1–5. [CrossRef]
15. Zhang, W.; Lin, D.; Zhang, H.; Chen, C.; Zhou, X. A Lightweight Anonymous Mutual Authentication with Key Agreement Protocol on ECC. In Proceedings of the IEEE Trustcom/BigDataSE/ICESS, Sydney, Australia, 1–4 August 2017; pp. 170–176. [CrossRef]
16. Certicom Research. *SEC 4: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV)*; Certicom Research: Mississauga, ON, Canada, 2013.
17. Cramer, R.; Shoup, V. Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack. *SIAM J. Comp.* **2003**, *33*, 167–226. [CrossRef]
18. Bjørstad, T.E.; Dent, A.W. Building Better Signcryption Schemes with Tag-KEMs. In Proceedings of the International Conference on Public Key Cryptography (PKC), New York, NY, USA, 24–26 April 2006; pp. 491–507. [CrossRef]
19. Bjørstad, T.E.; Dent, A.W.; Smart, N.P. Efficient KEMs with Partial Message Recovery. In Proceedings of the International Conference on Cryptography and Coding, London, UK, 12–14 December 2023; pp. 233–256. [CrossRef]
20. Barnes, R.; Bhargavan, K.; Lipp, B.; Wood, C. RFC9180-Hybrid Public Key Encryption. February 2022. Available online: https://www.rfc-editor.org/rfc/rfc9180.html (accessed on 2 February 2024).
21. Zheng, Y. Digital Signcryption or How to Achieve Cost(Signature & Encryption) « Cost(Signature) + Cost(Encryption). In Proceedings of the Annual International Cryptology Conference CRYPTO '97, Santa Barbara, CA, USA, 17–21 August 1997; Volume 1294, pp. 165–179.
22. Dent, A.W. Hybrid Cryptography. IACR Cryptology ePrint Archive, 2004. p. 210. Available online: https://eprint.iacr.org/2004/210 (accessed on 30 April 2024).
23. Dent, A.W. Hybrid Signcryption Schemes with Outsider Security. In Proceedings of the Information Security (ISC 2005), Singapore, 20–23 September 2005; pp. 203–217.
24. Dent, A.W. Hybrid Signcryption Schemes with Insider Security. In Proceedings of the Information Security and Privacy (ACISP), Brisbane, Australia, 4–6 July 2005; pp. 253–266.
25. Gorantla, M.C.; Boyd, C.; Nieto, G.; Manuel, J. On the Connection between Signcryption and One-Pass Key Establishment. In Proceedings of the Cryptography and Coding, Cirencester, UK, 18–20 December 2007; pp. 277–301.
26. Krawczyk, H. HMQV: A High-performance Secure Diffie-Hellman Protocol. In Proceedings of the Annual International Conference on Advances in Cryptology (CRYPTO), Santa Barbara, CA, USA, 14–18 August 2005; pp. 546–566.
27. Okamoto, T.; Tso, R.; Okamoto, E. One-Way and Two-Party Authenticated ID-Based Key Agreement Protocols Using Pairing. In Proceedings of the International Conference on Modeling Decisions for Artificial Intelligence (MDAI), Tsukuba, Japan, 25–27 July 2005; pp. 122–133. [CrossRef]
28. Challa, S.; Wazid, M.; Das, A.K.; Kumar, N.; Reddy, A.G.; Yoon, E.-J.; Yoo, K.-Y. Secure Signature-Based Authenticated Key Establishment Scheme for Future IoT Applications. *IEEE Access* **2017**, *5*, 3028–3043. [CrossRef]
29. Porambage, P.; Braeken, A.; Schmitt, C.; Gurtov, A.; Ylianttila, M.; Stiller, B. Group Key Establishment for Enabling Secure Multicast Communication in Wireless Sensor Networks Deployed for IoT Applications. *IEEE Access* **2015**, *3*, 1503–1511. [CrossRef]

30. Srinivas, J.; Das, A.K.; Wazid, M.; Vasilakos, A.V. Designing Secure User Authentication Protocol for Big Data Collection in IoT-Based Intelligent Transportation System. *IEEE Internet Things J.* **2021**, *8*, 7727–7744. [CrossRef]
31. Porambage, P.; Schmitt, C.; Kumar, P.; Gurtov, A.; Ylianttila, M. Two-phase Authentication Protocol for Wireless Sensor Networks in Distributed IoT Applications. In Proceedings of the 2014 IEEE Wireless Communications and Networking Conference (WCNC), Istanbul, Turkey, 6–9 April 2014; pp. 2728–2733. [CrossRef]
32. Pu, C.; Lim, S. A Light-Weight Countermeasure to Forwarding Misbehavior in Wireless Sensor Networks: Design, Analysis, and Evaluation. *IEEE Syst. J.* **2018**, *12*, 834–842. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

future internet

MDPI

Article

Conservative Interference Injection to Minimize Wi-Fi Sensing Privacy Risks and Bandwidth Loss

Aryan Sharma *, Haoming Wang, Deepak Mishra * and Aruna Seneviratne

School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, NSW 2052, Australia; haoming.wang@unsw.edu.au (H.W.); a.seneviratne@unsw.edu.au (A.S.)
* Correspondence: aryan.sharma@unsw.edu.au (A.S.); d.mishra@unsw.edu.au (D.M.)

Abstract: With the impending integration of sensing capabilities into new wireless standards such as 6G and 802.11 bf, there is a growing threat to public privacy. Recent studies have revealed that even small-scale activities, like keyboard typing, can be sensed by attackers using Wi-Fi Channel State Information (CSI) as these devices become more common in commercial spaces. This paper aims to model the minimum CSI data rate required to sense activities in the channel and quantifies the detection accuracy of WiFi-based keystroke recognition in relation to the CSI sensing data rate. Our experimental findings using commercial-off-the-shelf hardware suggest that interference can be used as a defence strategy to degrade the CSI data rate and prevent undesirable Wi-Fi sensing attacks. To achieve a reduced data rate, we propose an extension to Bianchi's model of CSMA/CA systems and establish a new mathematical relationship between channel contention and the available CSI. This proposed relationship was empirically verified, and our contention-based defence strategy was experimentally validated. Experiments show that our contention-based defence strategy increases the chances of evading undesired WiFi-based keystroke recognition by around 70%. By leveraging prior work that shows a degradation in CSI quality with lower transmission rates, we show that conservative interference injection can sufficiently reduce sensing accuracy whilst maintaining channel bandwidth.

Keywords: Wi-Fi sensing; privacy; machine learning; IoT

1. Introduction

1.1. Background

As low-cost Internet of Things (IoT) devices become increasingly pervasive, their coordination into sensor networks has enabled new applications in industrial and home settings. To this end, Wi-Fi is an essential IoT enabler, facilitating communication between these coordinated devices to allow for data sharing and user interfacing. However, more recently, a new paradigm in wireless sensing has demonstrated that the Wi-Fi signals themselves can be used to sense the physical environment [1,2]. It has been shown that artificial intelligence (AI) and machine learning (ML) can be applied to analyse Wi-Fi Channel State Information (CSI) in order to monitor environmental occupancy, movement and the presence of intruders [1]. These applications of Wi-Fi sensing have been extended towards more private human features such as hand movements [2] or keyboard inputs [3,4]. This is creating new security, privacy and trust issues in IoT networks as our cyber information is at risk. To restore the privacy of IoT users, there is growing attention towards the development of defence strategies to prevent unwanted Wi-Fi sensing [5–8]. In this endeavour, the use of adversarial interference has been beneficial for reducing sensing accuracy and

restoring user privacy as shown in Figure 1. Here, interference signals are injected (green) in order to impede the sensing signals (red). Unfortunately, the existing literature in this domain uses very high interference rates, which is a significant burden to the user in terms of bandwidth loss. Thus, there is a need for further work on interference-based defence techniques in order to improve their spectral efficiency.

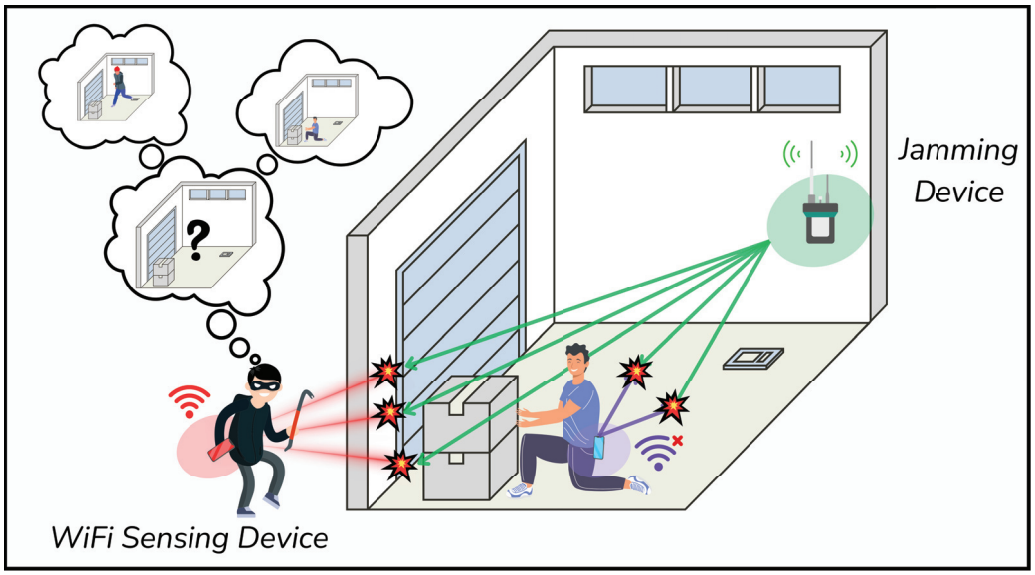


Figure 1. Interference topology.

1.2. Motivation and Contributions

Prior work on Wi-Fi sensing has demonstrated a growing threat to everyday civilians, as their gestures and personal information can be monitored by malicious attackers armed only with commercial Wi-Fi hardware. Hence, we are motivated to develop protection methods to thwart attempts to monitor unwilling patrons. We have identified some prior works that show that a brute forced jamming mechanism can diminish the attacker's threat; however, this approach entails severe degradation of the wireless communications link. This paper first experimentally validates the threat of human gesture recognition attacks with Wi-Fi sensing. We then present novel insights into a contention-based defence mechanism for thwarting said attacks. We make the following contributions:

1. In this study, we replicated keystroke recognition attacks from previous research and discovered new information on the minimum CSI data rates needed to identify and categorize keystrokes through Wi-Fi sensing.
2. After verifying that the accuracy of Wi-Fi sensing drops as a function of the CSI data rate, we extend Bianchi's [9] model of saturated CSMA/CA systems to create a mathematical relationship between the channel contention and the accessible CSI data rate.
3. We conducted an experimental validation of our mathematical model that describes the available CSI data rate in relation to the level of contention. Our findings show that as more devices access the Wi-Fi channel, the CSI data rate decreases, which implies that we may not have sufficient information to detect keystrokes.
4. Using this contention-based drop in the CSI rate, we motivate a defence strategy against Wi-Fi-based keystroke inference attacks. We achieve this by injecting controlled interference through another device that transmits ping packets to reduce the CSI sensing rate below the required threshold for accurate sensing.
5. We present a trade-off study that details the consequences of such contention-based defence strategies on the available Wi-Fi bandwidth for normal users.

The paper is structured as follows. In Section 2, we summarise state-of-the-art Wi-Fi sensing and its defence strategies. In Section 3, we provide a more detailed description of the Wi-Fi sensing system model and the importance of data rates for the system accuracy. We subsequently demonstrate a working test-bed of Wi-Fi-based keystroke detection in Section 4 and experimentally highlight the importance of CSI data rates. In Section 5, we propose a model to exploit Wi-Fi contention protocols to reduce CSI data rates in defensive endeavours. This contention model is then applied in Section 6 to thwart the accuracy of our keystroke detection test-bed. Finally, in Section 7, we provide insights on the trade-off between security improvement and channel bandwidth.

2. State of the Art

2.1. Channel State Information

Wi-Fi sensing is based on the propagation of radio waves through physical environments. As these signals travel from a Transmitter (Tx) to a Receiver (Rx), they encounter physical obstacles such as the air, walls, people and objects [10]. As the signal encounters these obstacles that cause reflection, refraction, diffraction and scattering, it splits into multiple paths that each arrive at the Rx at varying times (i.e., phase shift) with varying power (i.e., amplitude). This phenomenon is otherwise known as the multipath effect, and the overall multipath profile of a channel can be concisely characterised by a complex-valued metric known as Channel State Information (CSI) [1]. Due to the Orthogonal Frequency Division Multiplexing (OFDM) transmission scheme in 802.11n and subsequent Wi-Fi standards, the signal is modulated over a large domain of frequencies known as subcarriers [1]. CSI can be collected independently for each OFDM subcarrier, allowing us to analyse the propagation characteristics of the wireless environment as a function of frequency. In the same manner that varying wavelengths of visible light illuminate different objects, it has been shown that the CSI of different subcarriers can have utility for sensing different physical phenomena [11]. Tools such as Nexmon [12] sample CSI for all OFDM subcarriers on every incoming Wi-Fi packet. Hence, the data rate of CSI systems is bounded by the maximum number of packets they can propagate in the channel. CSI is then used to infer information about the channel.

2.2. Large-Scale Applications

Macroscopic changes in the channel, such as the introduction of new objects, cause large-scale changes to the multipath propagation characteristics of an environment. This is reflected in CSI, which can have large changes in the amplitude level or fluctuations over a captured time series [1,13]. To correlate changes in CSI with the environment, the CSI is first filtered to remove noise. Common filtering approaches include Hampel filters to remove outliers [1] and lowpass filters to smooth the CSI [14]. Following this, features can be calculated to correlate with changes in the physical environment. In applications such as object detection [11,15], the CSI is analysed on a frame-by-frame basis since the sensing subject is not time-varying. Hence, statistical features calculated on the set of CSI amplitudes have been used in conjunction with Support Vector Machines (SVMs) to predict the presence of different people or objects. Beyond this, large-scale changes in a physical environment such as occupancy levels have also been detected using absolute CSI amplitude levels. In addition to this, some large-scale sensing applications have used more complex temporal features calculated over the sampled time series of CSI. This includes feature calculation techniques such as Principle Component Analysis (PCA) [10] and Wavelet Transforms [1,10], which allow temporal variations to be analysed and subsequently fed into supervised machine learning (ML) algorithms to predict behaviours such as falling, walking and waving [16].

2.3. Fine-Grained Gesture Recognition

More recent work has applied advanced signal processing techniques to sense microscopic phenomena in the channel. To detect smaller changes in the channel, such as the movement of an arm [17], prominent works compute advanced features such as power distributions [18] or velocity distributions [17]. These higher-order features are typically applied with more complex learning algorithms such as Convolutional Neural Networks (CNNs) [19] or Long Short-Term Memory (LSTM) classifiers [1]. Of particular interest to us is the application of human keystroke recognition [3]. WiKey and other gesture recognition algorithms use moving window approaches to identify the start and end of some gesture event [3]. The identified windows contain waveforms of CSI amplitudes, which vary based on the gesture (i.e., a key press), and their algorithm subsequently uses Discrete Wavelet Transforms (DWTs) and Dynamic Time Warping (DTW) to distinguish between different waveforms. This approach is similar to generalised human activity recognition, which also seeks to classify based on the frequency composition of CSI [20].

2.4. Protection Methods

As evidenced by the success of prior work in fine-grained gesture recognition, there is a growing need for methods that secure privacy in the presence of Wi-Fi networks. Prior work on thwarting Wi-Fi sensing has introduced jamming signals into the wireless channel [5–7,21], causing a degradation in the sensing accuracy. In [5], interference was introduced on adjacent Wi-Fi channels in order to create competition for channel resources and reduce the CSI available to the sensing system. This subsequently caused a degradation in the sensing accuracy. Then, in [7], it was shown that the interference can be introduced in a time-varying manner in order to bias the measured CSI towards certain activity classes, thereby intentionally misleading the sensing system. This approach requires prior knowledge of the sensing systems ML parameters, which is unlikely. Thus, we focus on untargeted interference in order to degrade sensing accuracy. One aspect of interference injection that is overlooked is its impact on channel bandwidth. To the best of our knowledge, no prior work investigates the trade-off between improved security and loss in bandwidth due to interference injection. Thus, there is scope for more work to understand how the amount of interference can be chosen intelligently to preserve the communications channel for normal users.

3. System Description

In this section, we describe the working principles of Wi-Fi sensing and its dependence on the underlying data rate.

3.1. Wi-Fi Sensing System

As shown in Figure 2, Wi-Fi sensing is predicated on the propagation of signals (red) through a physical environment. As the signal encounters physical entities, its power and phase are altered, causing a disparity in the initial signal and that which is received by Rx. Where the physical entities are moving, such as the finger movements in Figure 2, the artefacts introduced in the wireless signal are also time varying as shown by the spikes in the signal (red) received by Rx. The relationship between the transmitted and received signals can be formally expressed as follows:

$$Y(f,t) = H(f,t)X(f,t) + N(f,t) \tag{1}$$

where the signal Y is received in continuous time t for all OFDM subcarrier frequencies f. This received signal is equal to the transmitted signal X after it incurs the multipath transformation, represented by H (CSI). The Gaussian white noise in CSI is represented by

N. Using this relationship, tools such as Nexmon [12] are able to take 'snapshots' of the CSI data (H) by inspecting the pilot bits of all received packets. The CSI data can be represented as $H[f,k]$ where $t = kT_s$. Here, we measure the CSI at k locations with a separation of T_s along the originally continuous domain t. The frequency domain representation of H is thus the following:

$$H[f,n] = \sum_{k=0}^{N-1} H[f,k]e^{-j\frac{2\pi}{N}nk} \tag{2}$$

where for all subcarriers f, $H[f,n]$ is the Discrete Fourier Transform (DFT) of the sequence $H[f,k]$. Equation (2) shows that the DFT representation of the underlying CSI is inherently dependent on the number of CSI snapshots used to compute the DFT, N, and the discrete domain variable k, which is itself dependent on the sampling period T_s. As described in Section 2, existing techniques in Wi-Fi sensing rely heavily on frequency analysis of CSI to discern gestures and movements of different speeds. Hence, we conclude that Wi-Fi sensing ML algorithms are highly dependent on the CSI sampling rate.

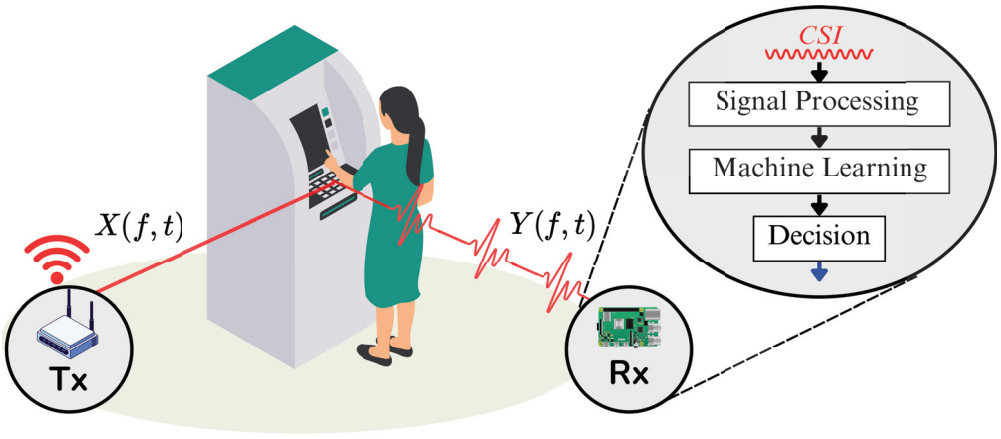


Figure 2. Wi-Fi sensing system overview.

3.2. CSI Sensing Rate and Coherence Time

Beyond the algorithmic dependence on CSI data rates, it is important to note that Wi-Fi sensing systems require a minimum data rate to effectively understand the changing wireless channel. To better understand this, we use the concept of channel coherence time. In order to sense activity in the channel, the CSI data must be captured faster than the rate of change of the channel. The time for which the channel is considered unchanging is known as coherence time, and it can be expressed in 2.4 GHz wireless channels as follows [22]:

$$T_{\text{coherence}} = \frac{9}{16\pi\frac{v}{\lambda}} \tag{3}$$

where v is the speed of movement in the channel (the activity being sensed), and λ is the signal wavelength (0.12 m for 2.4 GHz Wi-Fi). The required CSI data rate is then the reciprocal of the coherence time, and we compute this for several common sensing applications in Table 1.

Table 1. Approximated data rates for various sensing applications.

Activity	Speed (m/s)	$T_{\text{coherence}}$ (s)	Req. CSI Rate (Hz)
Walking	1.5	0.016	63
Running	2.7	0.008	129
Typing	4	0.005	189
Occupancy	0.7	0.031	33

In summary, the accuracy of Wi-Fi sensing is influenced by the CSI data rate, which is limited by the coherence time of the activity being sensed. Additionally, the hyperparameters of the ML algorithm are governed by the data rate for training. Therefore, any fluctuations in the rate of CSI capture will negatively impact the system's performance. To prove this assertion and determine the necessary CSI data rate, we will be conducting experiments on Wi-Fi-based keystroke recognition for varying transmission rates in the following section.

4. Keystroke Recognition Using Wi-Fi Sensing

In this section, we implement keystroke recognition using a state-of-the-art algorithm [3], before performing a subsequent experiment to show its reliance on the underlying data rate.

4.1. Keystroke Recognition Model

It has been shown in prior work [3] that a Wi-Fi sensing apparatus can be used to detect which key is pressed on a keyboard by computing features on CSI time series data. This flavour of the attack is particularly concerning since it extends the domain of privacy risks beyond physical attributes towards cyber information that may be entered through a computer keyboard. We implemented the algorithm initially proposed by WiKey as demonstrated by Figure 3 below.

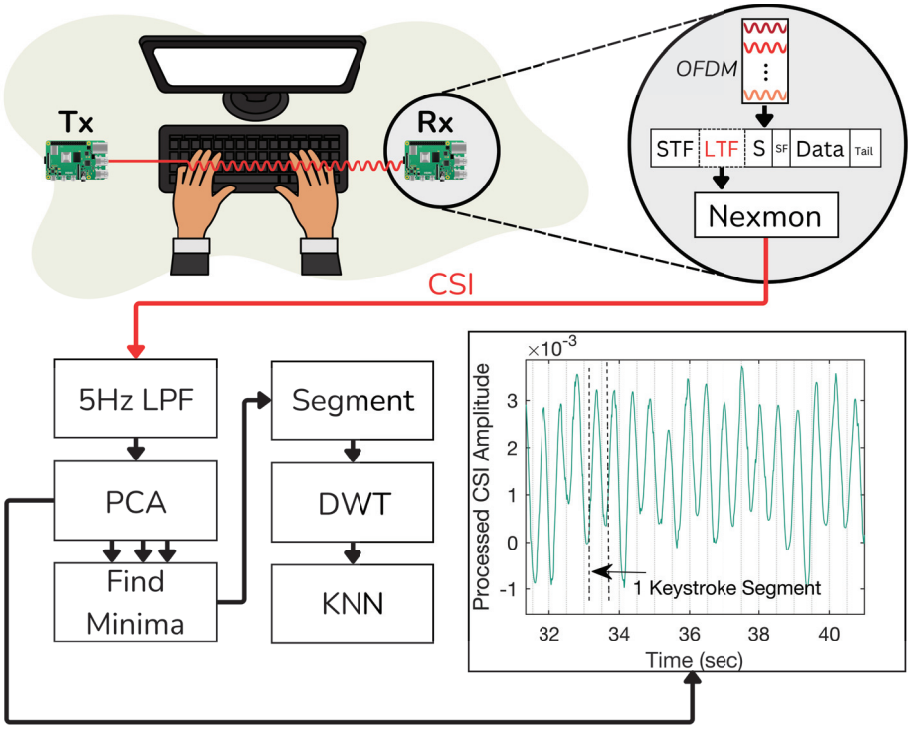


Figure 3. WiFi-based keystroke detection model.

In Figure 3, we illustrate the algorithm for CSI-based keystroke recognition. As before, Tx propagates a Wi-Fi signal (red) which incurs time-varying oscillations due to the typing hand. This time varying Wi-Fi signal is then received by Rx. The CSI time series is first processed with a lowpass filter to remove noise above the cutoff frequency of 5 Hz. Then, the 56 time series (one for each of the 56 OFDM subcarriers) are collectively processed via PCA. The 2nd, 3rd and 4th PCA vectors are used in parallel for all subsequent computations.

Encapsulated within Figure 3, we present a plot of a CSI time series after lowpass filtering and PCA. The CSI data was captured whilst a human repeatedly pressed the 'Q'

key on a keyboard. The x-axis represents time, and the y-axis represents the amplitude of the processed CSI. We observe clearly that the CSI amplitude varies in sync with the repeated keystrokes. To extract this keystroke information, we find the local minima in the time series to isolate each of the 'crevices' at the end of a keystroke. The samples between these minima are then considered to be keystrokes, and we perform segmentation of the time series to extract the detected keystrokes. Each segmented keystroke is then subject to the DWT, which characterises the shape of the keystroke segment by a frequency decomposition using Daubechies Wavelet 4. These shape features are then used as inputs to 3 KNN's, which utilise DTW distances to calculate the distance between points and their neighbours. The decisions from the 3 KNN algorithms are then multiplexed in order to make an overall classification of the keystroke.

4.2. Experimental Setup

In a standard office building within the range of commercially operating Wi-Fi networks, we setup our experimental apparatus as depicted below.

In Figure 4, we depict the experimental setup for human keystroke recognition using Wi-Fi CSI. We positioned a Wi-Fi Transmitter (Tx) on one side of a standard keyboard, which broadcasts ping packets at a rate of 1300 Hz. On the other side, we placed a Raspberry Pi 4B Receiver (Rx) equipped with the Nexmon CSI extraction firmware [12]. This is denoted by the red circle in Figure 4. This device is manufactured by Raspberry Pi Ltd. (Cambridge, UK), and was sourced online within Australia via. Element 14. The Nexmon firmware activates monitor mode on the Raspberry Pi 4B Network Interface Card (NIC) and then captures CSI for every received packet. Given that Tx is propagating packets at 1300 Hz, Rx will capture 1300 CSI snapshots per second. All communications occur on channel 2 of the 2.4 GHz Wi-Fi spectrum, with a bandwidth of 20 MHz and CSI sampled for 56 subcarriers. The experimental subject (pictured) pressed the set of keys Q, Z, H, M and P a total of 100 times in the training phase and 50 times in the testing phase. Without loss of generality, we chose this subset of keys since they are evenly spread across the keyboard and allow for simpler analysis compared to a full keyboard.

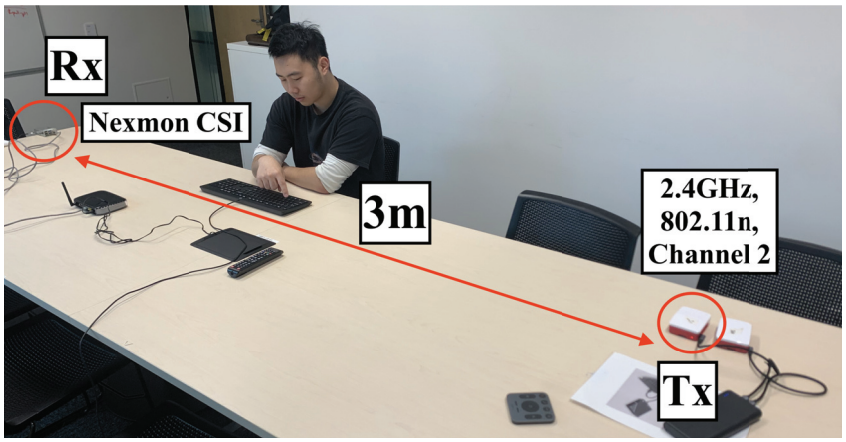


Figure 4. Keystroke recognition experimental setup.

4.3. Results

The experimental data was passed into the processing framework described above, and we obtained the following result.

In Figure 5, we present a confusion matrix of the ML algorithm when evaluated with our experimental data. The rows represent true classes, and the columns represent predicted classes. Hence, the leading diagonal represents the true positive rates for each key, with darker colors corresponding to higher rates. We observe that the classifier has an

overall true positive rate of 84.96%. The largest error occurs when classifying the H key being pressed, which is incorrectly predicted as M 19.20% of the time. The misclassification between these two classes can be explained by observing the relative proximity of the M and H keys on a keyboard. Furthermore, the worst performing class overall is the H key, with a true positive rate of 71%. This once more reflects the position of the H key, in between all the other keys. Overall, this result supports those achieved in prior work [3], demonstrating the threat posed to user keystroke information by Wi-Fi sensing.

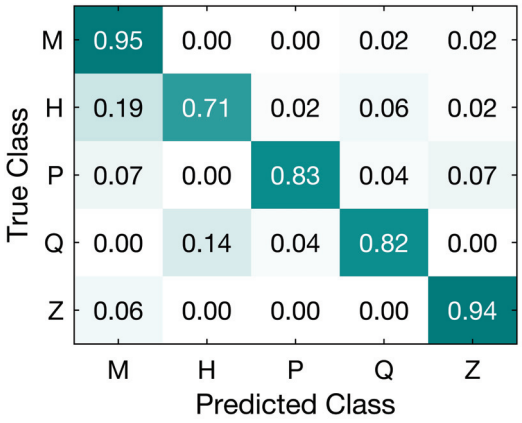


Figure 5. Classification accuracy.

4.4. Minimum CSI Sensing Rate for Reliable Sensing

The result in the previous section was achieved with a CSI data rate of 1300 Hz. To illustrate the importance of this rate, we perform a subsequent experiment in which we change the transmission frequency of the Wi-Fi sensing device (Tx). We setup the devices as in Figure 4 and asked the participant to press a key 50 times. This process is repeated for transmission frequencies varying between 1300 Hz and 20 Hz. We observe what percentage of the 50 keystrokes the system was able to detect from the captured CSI. This is presented in Figure 6.

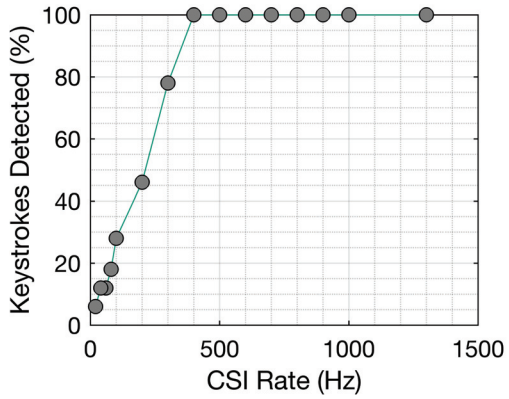


Figure 6. Impact of CSI Rate.

In Figure 6, we present a result that demonstrates the effect of the CSI data rate on the keystroke detection algorithm. The x-axis represents the data rate (Hz) of the sensing system, and the y-axis represents the percentage of 50 keystrokes that were detected from the CSI. Then, each plotted point represents an empirical result. Here, we note that *detection* refers to the ability of the system to extract a keystroke event from the CSI waveform, *not* the classification accuracy.

We observe that CSI data rates above 400 Hz are able to detect 100% of the keystrokes. Below this, we see a clear negative trend in the keystroke detection rate as a function of the

CSI data rate. A key point of interest is 200 Hz, below which only 47% of the keystrokes can be detected by the system. This corroborates our analytical finding in Table 1, where the critical data rate for keystroke recognition was 189 Hz. Given that we are discussing the ability to *detect* keystrokes, these results do not reflect ML performance or the reliance of feature calculation schemes on the sampling rate. Instead, this result demonstrates the importance of dense sampling to reconstruct human movement in sampled CSI time series. This issue is akin to the Nyquist–Shannon sampling problem for sampling any time varying information. Overall, the results in this section demonstrate the importance of the underlying CSI data rate for Wi-Fi sensing systems.

5. Proposed Contention Model for Defence Against Wi-Fi Sensing

The previous section showed empirically that the efficacy of WiFi-based keystroke recognition attacks is predicated on the underlying CSI data rate. In this section, we theorise how the CSI data rate can be reduced by intentionally injecting contention into the Wi-Fi channel. In this way, we formulate a contention-based model for preventing undesirable Wi-Fi sensing applications and safeguarding our privacy.

5.1. CSMA/CA in 802.11 Networks

802.11 networks are designed with Medium Access Control (MAC) protocols such as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) to handle a large amount of devices attempting to access the wireless channel resources simultaneously [23]. CSMA/CA requires stations to sense the channel to ensure that it is idle prior to attempting any transmission [24]. When a station is ready to send a packet, it listens to the channel for a period known as the Distributed Inter-Frame Space (DIFS) interval [24]. If the channel remains idle throughout the entire DIFS interval, the station backs off for a randomly chosen number of discrete time slots. This random back-off duration is uniformly distributed between $(0, W-1]$, where W is a variable known as the contention window [23–25]. In each slot where the station senses an empty channel, it decreases the back-off counter. Once the back-off counter reaches zero, the station can attempt to transmit again using the wireless medium. If a collision is detected, the station again randomly chooses a back-off interval, this time choosing an integer uniformly distributed between $(0, 2W-1]$, and the process repeats. The upper limit iteratively increases as a function of the number of collisions (m) [25] by $2^m W - 1$. This is limited to a maximum ceiling of $2^{m_{\max}} W - 1$, where $m_{\max}$ is the maximum number of collisions allowed, which varies between Wi-Fi hardware.

5.2. Throughput of CSMA/CA WLANs

The CSMA/CA process described above has been represented using a Markovian model in a prominent work by Bianchi around the time of its advent [9]. Subsequent work [26] has built upon Bianchi's model to express the collision probabilities of 802.11 stations. This can be boiled down to two important expressions:

$$p = 1 - (1 - \pi)^{N-1} \tag{4}$$

$$\pi = \frac{2(1-2p)}{(1-2p)(W+1) + pW(1-(2p)^m)} \tag{5}$$

where π is the probability that a station will transmit during a slot, and then p is the probability that there is a collision in that slot. The probability π is considered to be equal for all stations, operating under the assumption that they are all attempting to perpetually transmit packets. This is formally referred to as the saturation condition by Bianchi [9]. These two expressions can be solved simultaneously for π and p as a function of the number of stations (N). We use the following set of parameters for 802.11n networks:

Subsequently, the parameters in Table 2 can be used to solve Equations (4) and (5) simultaneously. We solve for $N = [2, 25]$ devices and illustrate the collision probabilities below.

Table 2. 802.11n CSMA/CA parameters.

Parameter	Value
Slot Time	20 µs
DIFS	50 µs
Short Inter-Frame Space (SIFS)	10 µs
Acknowledgement (ACK) Time	304 µs
T_{Header}	400 µs
T_{Payload}	3.4 ms
Minimum Contention Window (W)	16 slots

In Figure 7, we illustrate the collision probability (p) and the transmission probability (π) as functions of the number of competing stations in 802.11n networks. The x-axis represents the number of 802.11n devices in the network, and the y-axis represents probabilities ranging from 0 to 1. Then, the blue curve represents the probability of a collision occurring in the channel and the red curve represents the probability that any station will choose to transmit at a given time.

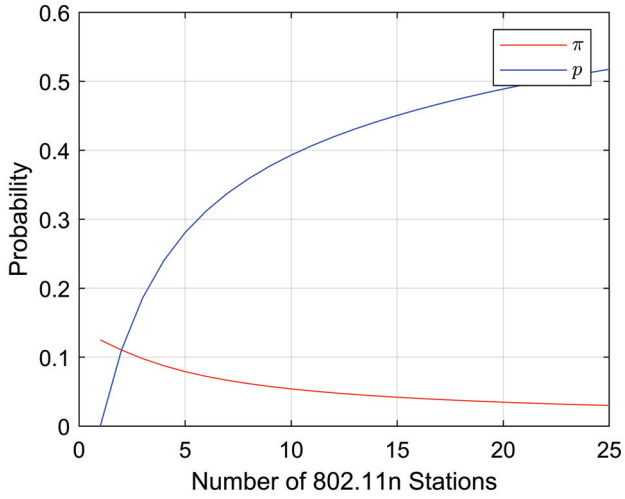


Figure 7. Collision probability of 802.11n networks with increasing users.

We observe that the probability of collisions within the channel grows logarithmically as more users enter the channel. This growth is particularly aggressive when the patronage increases from 2 to 5, with the probability of collision increasing from 0.11 to 0.28 (254% increase). We are subsequently interested in the effect this increased collision probability has on the overall throughput for all devices in the channel. Building on Equations (4) and (5), it has been further shown in prior work [26] that the throughput of a CSMA/CA network can be expressed as follows:

$$\tau = \frac{E[\text{Payload Transmitted in a slot}]}{E[\text{Duration of Slot Time}]}$$

$$\tau = \frac{P_s P_{tr} L}{P_s P_{tr} T_s + P_{tr}(1 - P_s) T_c + (1 - P_{tr}) T_{id}} \tag{6}$$

for a packet of length L and a slot time T_{id}. Here, the probability of transmission (P_{tr}) and probability of that transmission being successful (P_{s}) are further defined as follows:

$$P_{\text{tr}} = (1 - (1 - \pi)^N), P_s = \frac{N\pi(1-\pi)^{N-1}}{1-(1-\pi)^N} \tag{7}$$

and the average transmission time (T_{s}) and average time spent in collision (T_{c}) are expressed as follows:

$$T_s = T_{\text{Header}} + T_{\text{Payload}} + \text{SIFS} + \sigma + \text{ACK} + \text{DIFS} + \sigma \tag{8}$$

$$T_c = T_{\text{Header}} + T_{\text{Payload}} + \text{DIFS} + \sigma \tag{9}$$

where σ is the propagation delay. We solve Equation (6) for $N = [1, 10]$ devices, and vary the length of the payload between 700 bits (roughly the size of an ICMP ping packet) to 1900 bits.

In Figure 8, we illustrate the analytical throughput for 802.11n networks. The x-axis represents the number of Wi-Fi stations in proximity, and the y-axis represents the throughput (bits/sec) of a station. We illustrate 5 curves corresponding to the devices propagating packets of varying lengths.

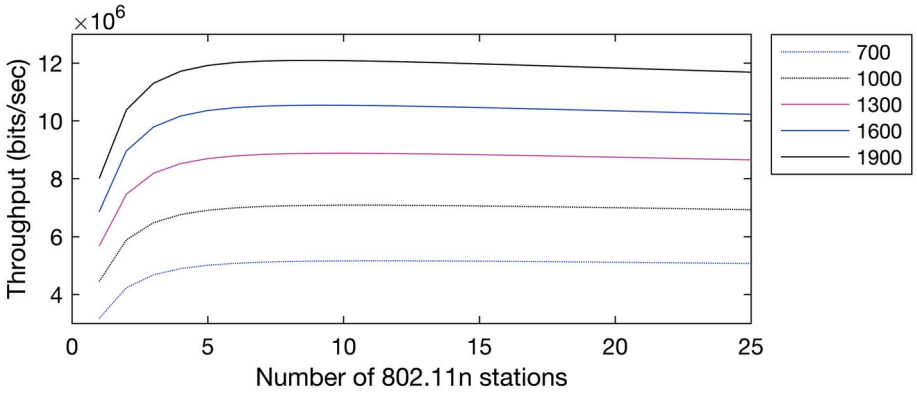


Figure 8. Throughput of 802.11n networks with increasing users and different packet lengths.

From the vertical spread of the various curves, we note that the throughput varies as a function of the packet size. This is symptomatic of the greater channel utilisation due to larger packet sizes since the stations spend less time in the inter-frame periods such as DIFS and SIFS. We further note that the decay in throughput appears to be independent of the size of the packets being transmitted. Overall, the total channel throughput decreases slightly as the number of stations increases, presumably due to the increased overhead of the distributed access protocol. This result represents the throughput in bits per second. We recall from Section 3 that CSI for Wi-Fi sensing is captured on a per-packet basis. Hence, we propose a new expression for the available CSI data rate as the quotient of throughput with the length of the transmitted packets, divided by the total number of stations as in Equation (10).

$$R_{\text{CSI}} = \frac{1}{N}\frac{\tau}{L} \tag{10}$$

We compute R_{CSI} for the throughputs derived earlier and plot the results below.

In Figure 9, we illustrate the CSI data rates for an 802.11n network with an increasing number of stations in the channel. The x-axis represents the number of stations transmitting in the channel, and the y-axis represents the effective CSI data rate of the channel. Each curve represents a simulation with packets of varying lengths. We note that a packet

length of 700 bits corresponds to ICMP ping packets, which are commonly used for Wi-Fi sensing applications.

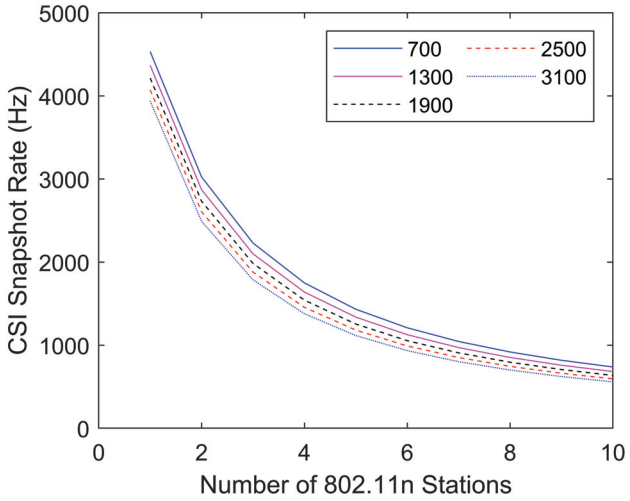


Figure 9. CSI snapshot rate under varying packet lengths.

We observe firstly that the CSI data rate reduces as the number of 802.11n users increases. This is initially a sharp decrease by a factor of 0.5 when the usage increases from one to two users, and the rate of decrease slows down subsequently. We further note that the CSI data rate is maximised for the smallest packet lengths, with the solid blue curve representing the highest CSI data rate for ping packets. This result reflects the smaller time spent occupying the channel per packet, and hence, more packets can use the channel resulting in a higher CSI data rate. Overall, the analytical results in this section support the notion that interference from other devices can reduce the effective CSI data rate by virtue of the 802.11n medium access control.

5.3. Empirical Validation

To validate our proposed theoretical model, we conduct an experiment to demonstrate how an increase in channel users causes a reduction in the available CSI data rate.

5.3.1. Experimental Setup

The experimental setup consists of a CSI sensing apparatus surrounded by interference devices as illustrated in Figure 10.

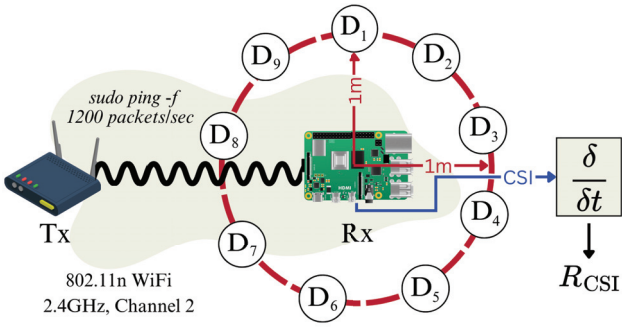


Figure 10. Experimental setup for validating analytical model.

As shown by the red arrows, the CSI sensing apparatus is surrounded by 10 interference devices at 1 m distance from Rx. In each trial, we collect CSI by transmitting Wi-Fi signals from Tx to Rx, as shown by the black wavy line, for a duration of 60 s, and we compute the average gradient ($\frac{d_{\text{CSI}}}{d_t}$) of the time series to estimate the average CSI data

rate. We first conduct a trial with none of the interference devices (D_n) operating, and we conduct 9 subsequent trials with each interference device iteratively switched on. When switched on, the interference devices propagate ping packets at the saturation rate using the `sudo ping -f` command.

5.3.2. Results

The results of the experiment are illustrated below.

In Figure 11, we present a plot of the CSI data rate that Rx was able to achieve with varying levels of contention in the channel. The x-axis represents the number of 802.11n stations sending ping packets, and the y-axis represents the average CSI data rate that Rx was able to achieve over the duration of 60 s. The blue line represents the analytical result from earlier, and the red line is the experimentally derived result.

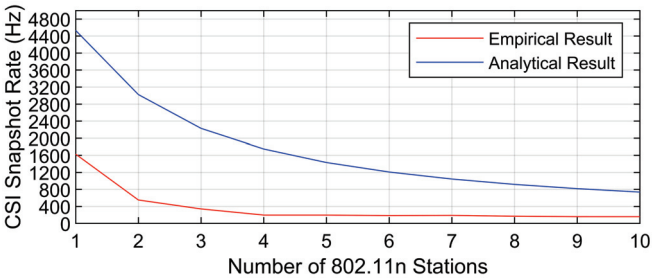


Figure 11. CSI snapshot rate with varying number of stations.

We observe a striking similarity in the shape of the empirical result with the analytical result. The CSI snapshot rate is initially 1200 Hz, and it decreases to less than 400 Hz with the addition of only a single interference device. Introducing subsequent devices causes further degradation in the achievable CSI snapshot rate to less than 200 Hz. Both curves have similar shapes, with the key difference being a vertical translation. This difference can be understood when considering the real-world limitations of 802.11 communications. Although the analytical result implies that a CSI data rate of 4400 Hz is achievable with a channel bit-rate of 72 Mbps, the actual bandwidth is limited by the computation power of the wireless devices and their Network Interface Cards (NICs). Overall, this result supports the theoretical model for contention-based defence against undesired Wi-Fi sensing attacks.

6. Application of Defence Model Against Keystroke Attacks

The previous section demonstrated analytically and empirically that channel contention can reduce the available CSI data rate. We utilise these ideas to develop a defence mechanism to thwart the efficacy of Wi-Fi-based keystroke recognition.

6.1. Experimental Setup

We utilise the same experimental apparatus as before, with the addition of an interference device as depicted in Figure 12.

The interference device is a Macbook Pro laptop on the same communications channel (Channel 2 of the 2.4 GHz Wi-Fi spectrum). It is configured to propagate ping packets at varying rates between 1300 Hz and 0 Hz, whilst the keyboard sensing apparatus operates with Tx propagating ping packets at 1300 Hz. As demonstrated by the ellipses in Figure 12, the proximity of these devices causes the range of their communications to overlap, resulting in competition for control over the channel resources. As illustrated by the wavy red signal in Figure 12, movement of fingers on the keyboard causes time varying changes in the Wi-Fi signal which are classified by the keystroke sensing system. To investigate how the varying interference packet rate influences the accuracy of the keystroke sensing

system, we once more collected CSI data during 100 keystroke events for each of the letters Q, Z, H, P and M and used the model in Section 4 to train a sensing system. The system was then evaluated with a dataset of CSI collected during 50 keystrokes for each of the same keys. We collected 6 such sets of testing data, under varying rates of contention from the laptop.

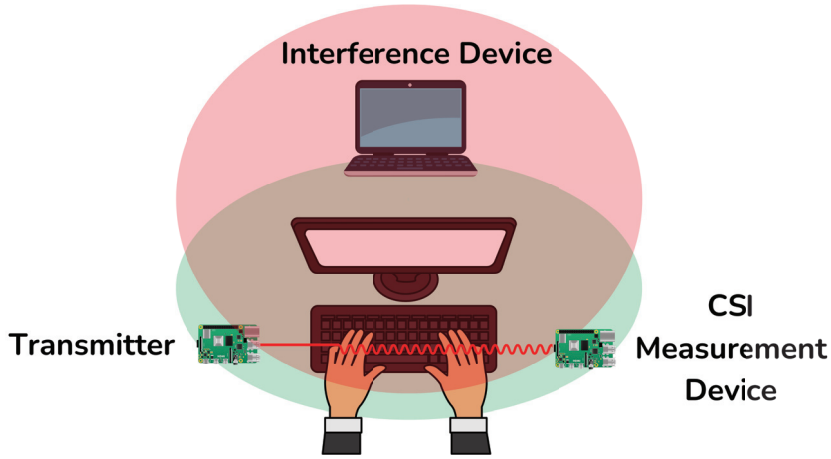


Figure 12. Setup for interference experiment.

6.2. Results

As the laptop begins to transmit ping packets, the captured CSI waveforms degrade, as demonstrated by Figures 13–17.

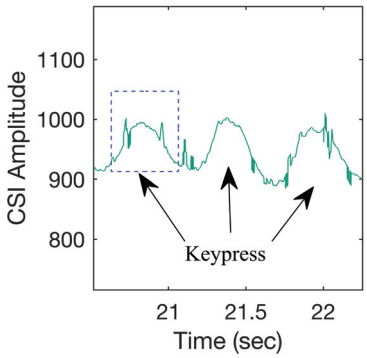


Figure 13. CSI time series with no interference.

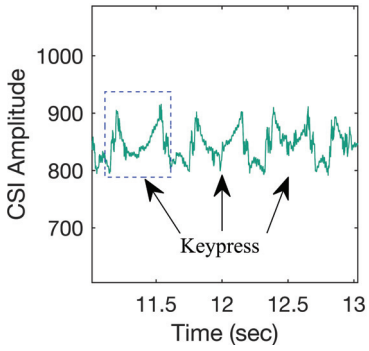


Figure 14. CSI time series with 50 Hz interference.

We observe firstly from Figure 13 that the keypress events are clearly visible in the sampled CSI as time-varying ripples, in line with our expectations as discussed in Section 3 and in line with our earlier results in Section 4. In each Figure, a single keystroke is outlined by the dotted box, and the arrows point towards the other observed keystrokes. Then, in Figures 14–17, there is an incremental degradation in the quality of the CSI and the

clarity of the keystroke events. In Figures 14 and 15, we are still able to discern three clear keypress events; however, in Figures 16 and 17, this is no longer possible. This gradual degradation of the CSI time series further supports our idea that maximum interference rates are not necessary, and lower interference can be applied to sufficiently degrade sensing. To understand the impact of the degraded CSI on the sensing accuracy, we proceed with evaluating the sensing output of the keystroke recognition threat model using this dataset.

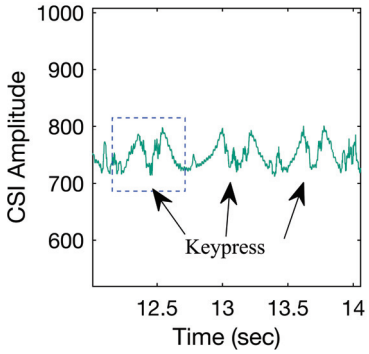


Figure 15. CSI time series with 250 Hz interference.

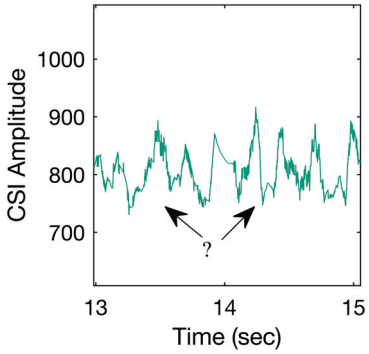


Figure 16. CSI time series with 500 Hz interference.

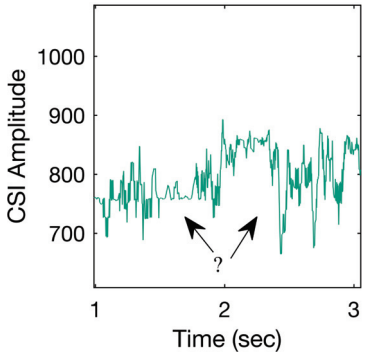


Figure 17. CSI time series with 1 kHz interference.

In Figure 18, we present the sensing accuracy against the interference packet rate. The x-axis represents the packet rate from the interfering laptop. The green left y-axis then represents the accuracy of the keystroke recognition system as it operates in the presence of said interference, and the blue right y-axis represents the CSI data rate under each level of interference. For each interference rate, the blue point thus denotes the achieved CSI data rate (Hz) and the green diamond denotes the achieved sensing accuracy. There is a clear negative correlation between the interference frequency and the sensing accuracy. With an initial interference rate of 50 Hz, the sensing accuracy drops from 85% to 70%. Beyond this, the sensing accuracy crosses below 50% at an interference rate of 500 Hz, corresponding to a CSI data rate of 700 Hz. This corroborates our observations from

Figures 13–17, where the keystrokes could no longer be clearly identified with interference rates of 500 Hz. Beyond this, the accuracy appears to decay in a decreasing manner asymptotically. The poor performance of the sensing system with a CSI Data Rate of 700 Hz due to the interference injected at 500 Hz is an interesting result. We recall the earlier result in Figure 6, where a CSI sensing rate of 700 Hz was able to yield a keystroke detection rate of 100%. Once more we reiterate that detection refers to the ability to pick up the keystroke event in the CSI time series, *not* the classification accuracy. Despite being able to pick up all the keystrokes, the sensing classification accuracy is below 50%. This result corroborates the ideas proposed in Section 3, where we postulated that the dependence of Wi-Fi sensing systems on frequency-based feature engineering techniques creates a susceptibility to changes in the CSI data rate. Even though all the keystrokes are picked up within CSI, the change in CSI sampling rate distorts the DWT features and hence reduces the accuracy of the sensing system. Overall, this experiment clearly shows that minimal interference can still be effectively used to defend against Wi-Fi sensing. To further visualise the improvement in privacy, we define the *Privacy Preserving Probability (PPP)* metric as the percentage probability of safeguarding a key from being detected by Wi-Fi sensing. This is calculated as the complement of the true positive detection rate for each key. We plot the PPP under several rates of defensive interference in Figure 19. The x-axis represents the five keys that the Wi-Fi sensing system is attempting to detect. Then, the height of each column represents the PPP with different levels of defensive interference. We observe that the PPP generally increases as a function of defensive interference. Generally, the PPP is higher for all keys when comparing 500 Hz interference with 0 Hz interference, and higher again when comparing 1500 Hz interference. This is congruent with our results in Figures 13–17, where increasing amounts of interference resulted in more degraded CSI waveforms. With 1500 Hz of interference, we can safeguard all keys with an overall PPP of 85%, an improvement of 70% over the baseline. Compared with prior work, the result with 500 Hz of interference demonstrates that lesser amounts of interference can considerably improve user privacy. One aspect of this result which is unclear is how to choose an appropriate level of interference. At the outset of this paper, we discussed prior work that belligerently injects dense packet streams to maximally thwart CSI. In lieu of this, our goal is to be more conservative in our application of interference to sufficiently impede sensing accuracy without overconsuming channel resources.

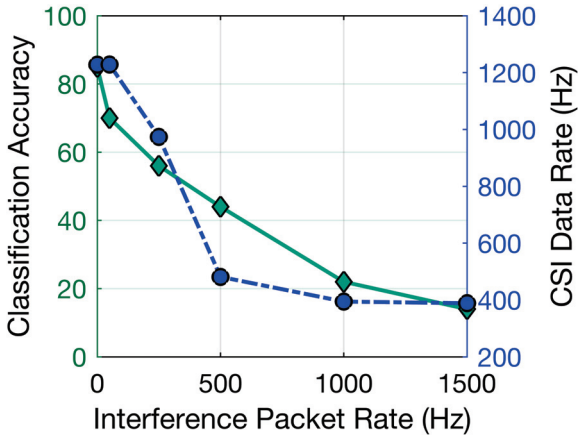


Figure 18. Effect of interference on Wi-Fi sensing accuracy.

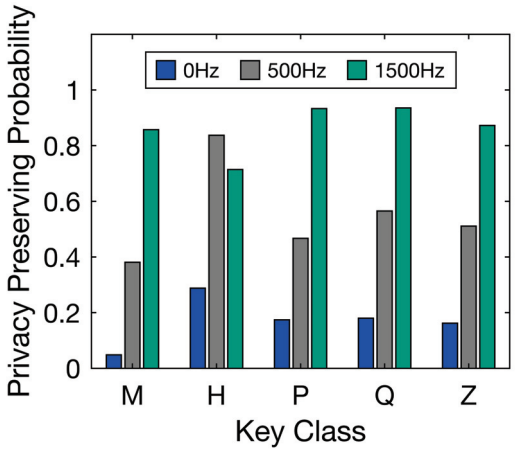


Figure 19. Security rate for each key as interference increases.

7. Tradeoff Between Privacy and Utility

The previous section showed that increasing amounts of interference increasingly degrade Wi-Fi sensing accuracy. This poses a question for IoT users who intend to use interference to fend off attackers, as they need to determine the optimal amount of interference that degrades attack accuracy while still allowing normal IoT communications. This section accordingly presents experiments to determine how the level of interference affects the usable Wi-Fi bandwidth (BW).

7.1. Experimental Setup

To evaluate the effect of interference defence on Wi-Fi bandwidth, we introduce a server–client pair to the experimental environment. Utilising the same setup illustrated in Figure 12, we position a Transmitter (Tx) in the channel, which propagates ping packets at a rate of 1000 Hz to a CSI measurement device. Meanwhile, an interference device is positioned adjacent to Tx. The independent variable is the rate of interference packet propagation, which we vary between 0 Hz and 1000 Hz. Concurrently, the server-client pair on the same Wi-Fi channel (2.4 GHz, Channel 2) transfers a 200 MB file over the wireless link. The dependent variable is the time taken for this file to transfer from the client to the server, which we subsequently convert to a bandwidth in Mbps.

7.2. Empirical Results

In this section, we compare the communications throughput under each interference level.

Figures 20 and 21 demonstrate the effect of interference on channel throughput. In Figure 20, the Wi-Fi throughput is plotted against the interference rates due to injected ping packets. A clear negative correlation between interference rate and Wi-Fi throughput is observed, as the linear fit has an RMSE of 0.4318 and R^2 of 0.952. Overall, an interference rate of 1000 Hz causes a large 25% reduction in throughput for the client–server pair. This once more motivates the use of more conservative interference injection to minimise throughput loss. The linearity of this trend is an interesting result, and can be compared against Figure 11. In Figure 11, the achieved CSI rate, which is proportional to throughput as per Equation (10), approaches an asymptote with the introduction of more devices. Due to hardware constraints on the Raspberry Pi 4B, we were unable to investigate whether the linear trend in Figure 20 approaches an asymptote with further increases in the transmission rate. We motivate future work to investigate this, when commodity devices are able to propagate more dense packet streams. To visualise the trade-off between the security and channel utility, Figure 21 plots the *PPP gain* against the measured throughput loss.

Here, the *PPP gain* is the percentage increase in the PPP over the baseline scenario with no interference. We observe a clear trade-off between the PPP gain and the available throughput. To achieve a PPP gain of 60%, we sacrificed nearly 25% of the available Wi-Fi throughput. Initially, we observe that a 20% gain in the PPP can be achieved with only 4% loss of throughput. This corresponds to the case of interference at a rate of 200 Hz. Referring back to Figure 18, this conservative interference rate is sufficient to reduce the keystroke sensing accuracy below 60%. The trade-off becomes more severe beyond this, as shown by the increasing slope of the curve in Figure 21. A further 20% gain in the PPP (from 20% to 40%) entails a much larger throughput loss of 11%. These results demonstrate the effectiveness of conservative interference injection in significantly mitigating sensing threats, while having minimal impact on Wi-Fi throughput.

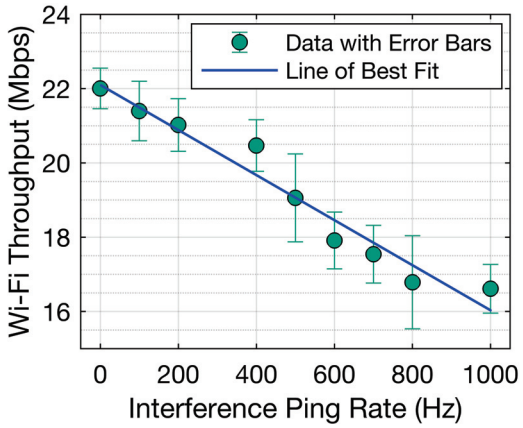


Figure 20. BW vs. interference.

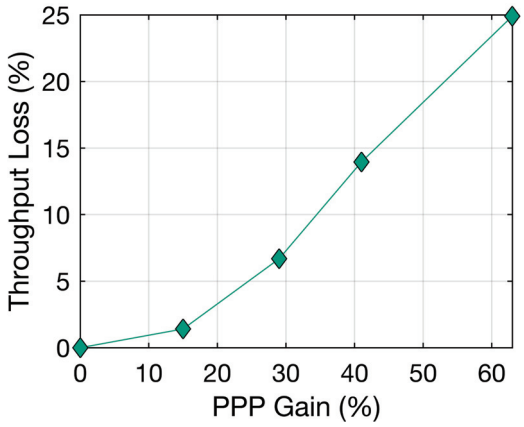


Figure 21. Privacy gains vs. BW.

8. Concluding Remarks

This paper introduces a conservative interference injection strategy to counteract privacy and security threats from Wi-Fi sensing. Through a survey of prior work in Wi-Fi sensing as well as the construction of our own testbed, we demonstrate that fine-grained human activities are under threat of being monitored with high accuracy. By experimental analysis using commodity devices, we demonstrate that this high accuracy of Wi-Fi sensing is highly dependent on the CSI data rate. We subsequently showed analytically and experimentally that this CSI data rate can be intentionally disrupted by other devices in the wireless channel. This principle is the basis of prior work in the state-of-the-art that utilises wireless interference to reduce the CSI rate and diminish sensing accuracy, thereby protecting user privacy. This paper further investigates this, revealing a positive correlation between the degree of interference introduced and the degradation in sensing

accuracy. This establishes a trade-off between achieving increased privacy and maintaining acceptable channel throughput for other Wi-Fi users. Compared to state-of-the-art approaches, we can thwart sensing applications with comparable efficacy at a fraction of the cost to channel bandwidth by injecting interference conservatively. Although our approach is applied specifically to keystroke recognition, the results are generalizable to other Wi-Fi-based gesture recognition systems, where interference can degrade the resolution of environmental movement. We suggest that future work should investigate the suitability of this conservative interference injection scheme in commercial settings under interference from additional devices. In such environments, the consumption of channel resources by other devices will create uncertainty around our ability to finely control the CSI rate of a sensing system. Furthermore, the increased environmental noise and poor Signal-to-Noise Ratio (SNR) of our interference devices may impede our ability to generate strong contention in the channel. Finally, whilst the methods and results in this paper are generalisable across Wi-Fi sensing applications, we motivate future work to apply it against coarse-grained sensing applications such as movement recognition. Future work should investigate whether these coarse-grained sensing applications, with relatively simple ML decision algorithms, are as sensitive to the CSI data rate and adversarial interference.

Author Contributions: Conceptualization, All; methodology, All; software, A.S. (Aryan Sharma) and H.W.; validation, A.S. (Aryan Sharma) and D.M.; formal analysis, All; investigation, All; resources, All; data curation, A.S. (Aryan Sharma) and H.W.; writing—original draft preparation, A.S. (Aryan Sharma) and H.W. and D.M.; writing—review and editing, D.M. and A.S. (Aruna Seneviratne); visualization, All; supervision, D.M. and A.S. (Aruna Seneviratne); project administration, D.M. and A.S. (Aruna Seneviratne); funding acquisition, D.M. and A.S. (Aruna Seneviratne). All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in parts by the Australian Research Council's Discovery Early Career Researcher Award (DECRA) under Grant number DE230101391 and by the Australian National Intelligence and Security Discovery Research Grants (NISDRG) under Grant number NI240100147. Aryan Sharma's participation is supported by the Cyber Security Cooperative Research Centre Limited, whose activities are partially funded by the Australian Government's Cooperative Research Centres Programme.

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Chen, C.; Zhou, G.; Lin, Y. Cross-Domain WiFi Sensing with Channel State Information: A Survey. *ACM Comput. Surv.* **2023**, *55*, 1–37. [CrossRef]
2. Yang, M.; Zhu, H.; Zhu, R.; Wu, F.; Yin, L.; Yang, Y. WiTransformer: A Novel Robust Gesture Recognition Sensing Model with WiFi. *Sensors* **2023**, *23*, 2612. [CrossRef] [PubMed]
3. Ali, K.; Liu, A.X.; Wang, W.; Shahzad, M. Recognizing Keystrokes Using WiFi Devices. *IEEE J. Sel. Areas Commun.* **2017**, *35*, 1175–1190. [CrossRef]
4. Li, M.; Meng, Y.; Liu, J.; Zhu, H.; Liang, X.; Liu, Y.; Ruan, N. When CSI Meets Public WiFi: Inferring Your Mobile Phone Password via WiFi Signals. In Proceedings of the ACM SIGSAC, Vienna, Austria, 24–28 October 2016; pp. 1068–1079. [CrossRef]
5. Zhang, J.; Li, M.; Tang, Z.; Gong, X.; Wang, W.; Fang, D.; Wang, Z. Defeat Your Enemy Hiding behind Public WiFi: WiGuard Can Protect Your Sensitive Information from CSI-Based Attack. *Appl. Sci.* **2018**, *8*, 515. [CrossRef]
6. Zhu, Y.; Xiao, Z.; Chen, Y.; Li, Z.; Liu, M.; Zhao, B.Y.; Zheng, H. Et Tu Alexa? When Commodity WiFi Devices Turn into Adversarial Motion Sensors. In Proceedings of the Network and Distributed Systems Security (NDSS) Symposium 2020, San Diego, CA, USA, 23–26 February 2020. [CrossRef]

7. Liu, J.; He, Y.; Xiao, C.; Han, J.; Ren, K. Time to Think the Security of WiFi-Based Behavior Recognition Systems. *IEEE Trans. Dependable Secur. Comput.* **2024**, *21*, 449–462. [CrossRef]
8. Huang, P.; Zhang, X.; Yu, S.; Guo, L. IS-WARS: Intelligent and Stealthy Adversarial Attack to Wi-Fi-Based Human Activity Recognition Systems. *IEEE Trans. Dependable Secur. Comput.* **2022**, *19*, 3899–3912. [CrossRef]
9. Bianchi, G. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE J. Sel. Areas Commun.* **2000**, *18*, 535–547. [CrossRef]
10. Wang, Z.; Huang, Z.; Zhang, C.; Dou, W.; Guo, Y.; Chen, D. CSI-based human sensing using model-based approaches: A survey. *J. Comput. Des. Eng.* **2021**, *8*, 510–523. [CrossRef]
11. Sharma, A.; Mishra, D.; Zia, T.; Seneviratne, A. A Novel Approach to Channel Profiling Using the Frequency Selectiveness of WiFi CSI Samples. In Proceedings of the 2020 IEEE Global Communications Conference (IEEE GLOBECOM), Taipei, Taiwan, 7–11 December 2020; pp. 1–6. [CrossRef]
12. Gringoli, F.; Schulz, M.; Link, J.; Hollick, M. Free Your CSI: A Channel State Information Extraction Platform For Modern Wi-Fi Chipsets. In Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization, Los Cabos, Mexico, 25 October 2019; pp. 21–28. [CrossRef]
13. Ma, Y.; Zhou, G.; Wang, S. WiFi Sensing with Channel State Information: A Survey. *ACM Comput. Surv.* **2019**, *52*, 1–36. [CrossRef]
14. Hernandez, S.M.; Bulut, E. WiFi Sensing on the Edge: Signal Processing Techniques and Challenges for Real-World Systems. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 46–76. [CrossRef]
15. Wang, F.; Han, J.; Lin, F.; Ren, K. WiPIN: Operation-Free Passive Person Identification Using Wi-Fi Signals. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Big Island, HI, USA, 9–13 December 2019; pp. 1–6. [CrossRef]
16. Ge, Y.; Taha, A.; Shah, S.A.; Dashtipour, K.; Zhu, S.; Cooper, J.; Abbasi, Q.H.; Imran, M.A. Contactless WiFi Sensing and Monitoring for Future Healthcare - Emerging Trends, Challenges, and Opportunities. *IEEE Rev. Biomed. Eng.* **2023**, *16*, 171–191. [CrossRef] [PubMed]
17. Zhang, Y.; Zheng, Y.; Qian, K.; Zhang, G.; Liu, Y.; Wu, C.; Yang, Z. Widar3.0: Zero-Effort Cross-Domain Gesture Recognition With Wi-Fi. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 8671–8688. [CrossRef] [PubMed]
18. Wang, W.; Liu, A.X.; Shahzad, M.; Ling, K.; Lu, S. Device-Free Human Activity Recognition Using Commercial WiFi Devices. *IEEE J. Sel. Areas Commun.* **2017**, *35*, 1118–1131. [CrossRef]
19. Sharma, A.; Jiang, W.; Mishra, D.; Jha, S.; Seneviratne, A. Optimised CNN for Human Counting Using Spectrograms of Probabilistic WiFi CSI. In Proceedings of the GLOBECOM 2022—2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 4–8 December 2022; pp. 1–6. [CrossRef]
20. Wang, Z.; Jiang, K.; Hou, Y.; Dou, W.; Zhang, C.; Huang, Z.; Guo, Y. A Survey on Human Behavior Recognition Using Channel State Information. *IEEE Access* **2019**, *7*, 155986–156024. [CrossRef]
21. Li, J.; Mishra, D.; Seneviratne, A. CSI-Based NTC Using Ambient WiFi: Opportunities and Challenges. In Proceedings of the 2020 IEEE Globecom Workshops (GC Wkshps), Taipei, Taiwan, 7–11 December 2020; pp. 1–6. [CrossRef]
22. Xiong, J.; Jamieson, K. SecureArray: Improving Wifi Security with Fine-Grained Physical-Layer Information. In Proceedings of the 19th Annual International Conference on Mobile Computing & Networking, Miami, FL, USA, 30 September–4 October 2013; ACM: New York, NY, USA, 2013; pp. 441–452. [CrossRef]
23. Tay, Y.C.; Chua, K.C. A Capacity Analysis for the IEEE 802.11 MAC Protocol. *Wirel. Netw.* **2001**, *7*, 159–171. [CrossRef]
24. Ziouva, E.; Antonakopoulos, T. CSMA/CA performance under high traffic conditions: Throughput and delay analysis. *Comput. Commun.* **2002**, *25*, 313–321. [CrossRef]
25. Chen, Y.; Agrawal, D.P. Effect of Contention Window on the performance of IEEE 802.11 WLANs. In Proceedings of the 3rd Annual Mediterranean Ad Hoc Networking Workshop, Bodrum, Turkey, 27–30 June 2004; Citeseer: University Park, PA, USA, 2004; pp. 27–30.
26. Manshaei, M.H.; Hubaux, J.P. Performance Analysis of the IEEE 802.11 Distributed Coordination Function: Bianchi Model. Mobile Networks. 2007. Available online: http://www.manshaei.org/files/C1-80211-Perf-Bianchi.pdf (accessed on 15 July 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

MDPI AG
Grosspeteranlage 5
4052 Basel
Switzerland
Tel.: +41 61 683 77 34

Future Internet Editorial Office
E-mail: futureinternet@mdpi.com
www.mdpi.com/journal/futureinternet



Disclaimer/Publisher's Note: The title and front matter of this reprint are at the discretion of the Guest Editors. The publisher is not responsible for their content or any associated concerns. The statements, opinions and data contained in all individual articles are solely those of the individual Editors and contributors and not of MDPI. MDPI disclaims responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.





Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-9064-4