REVIEW ARTICLE

IoT Security through ML/DL: Software Engineering Challenges and Directions

Haroon Arif[1], **Abdul Karim Sajid Ali**[1,*] **and Hussain Abdul Nabi**[2]

[1] College of Computing, Illinois Institute of Technology, Chicago, IL 60616, United States
[2] Superior University, Lahore 54000, Pakistan

Abstract

The Internet of Things (IoT) is increasingly integrated into modern software-driven systems across consumer, industrial, and healthcare domains. The heterogeneity of IoT devices, combined with their resource constraints, often renders conventional software security mechanisms insufficient, exposing systems to breaches and exploitation. This study examines recent IoT security incidents to illustrate common vulnerabilities in software-intensive IoT ecosystems, highlighting the resulting risks to critical applications. In response, we review emerging machine learning (ML)-driven security modules and deep learning (DL)-based intrusion detection software, positioning them as adaptive components that can be integrated into IoT system architectures. This review highlights recent peer-reviewed contributions, ensuring alignment with the most current developments in IoT security using ML and DL, and follows a systematic review methodology based on IEEE Xplore (2020–2024). The study further identifies software engineering challenges in integrating these intelligent modules into resource-constrained IoT environments and outlines future directions for building secure-by-design, AI-driven IoT software frameworks. Results demonstrate that ML- and DL-enhanced security modules strengthen software resilience by enabling real-time detection of cyber-attacks, reducing false alarms, and adapting to evolving threat landscapes. The review is structured to first discuss notable case studies of IoT security breaches, followed by an analysis of ML- and DL-based security modules, a comparative evaluation of their effectiveness, and finally, a discussion of key challenges and future research opportunities.

Keywords: internet of things (IoT), cybersecurity, machine learning (ML), deep learning (DL), intrusion detection system (IDS), anomaly detection, IoT security, adversarial attacks.

1 Introduction

The Internet of Things (IoT) is revolutionizing various sectors as a software-intensive ecosystem, where firmware on devices, application-layer protocols, and cloud-based platforms collectively drive automation and connectivity. However, this expansion faces critical security challenges because IoT software

Submitted: 11 August 2025
Accepted: 20 September 2025
Published: 02 November 2025

Vol. 1, **No.** 2, 2025.
10.62762/JSE.2025.372865

***Corresponding author:**
✉ Abdul Karim Sajid Ali
aali62@hawk.iit.edu

Citation
Arif, H., Ali, A. K. S., & Nabi, H. A. (2025). IoT Security through ML/DL: Software Engineering Challenges and Directions. *ICCK Journal of Software Engineering*, 1(2), 90–108.

© 2025 by the Authors. Published by Institute of Central Computation and Knowledge. This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

components are often resource-constrained, limiting the implementation of advanced security features. In addition, the lack of standardized protocols across heterogeneous software stacks exposes vulnerabilities that may result in data breaches, unauthorized access, and risks to critical infrastructure [1].

From a software engineering perspective, an IoT security framework extends beyond hardware layers to encompass a multi-layered software architecture that integrates application software, edge computing middleware, network protocols, and device firmware. Each layer requires tailored security mechanisms to ensure confidentiality, integrity, and availability of data [2]. A general IoT framework comprises an application layer, cloud layer, edge layer, network layer, and perception layer. Each of these layers can be viewed through a software engineering lens, where distinct software components assume specific security responsibilities. At the perception layer, device firmware and embedded operating systems must enforce secure booting and encryption of sensor data. In the network layer, communication software stacks such as TCP/IP and MQTT require safeguards against packet manipulation and denial-of-service attacks. At the edge and cloud layers, middleware and cloud APIs manage secure authentication, data aggregation, and encrypted transmission between distributed nodes. Finally, the application layer, which includes user-facing IoT software platforms, must implement secure access controls and privacy-preserving mechanisms. Figure 1 illustrates this layered IoT software framework, emphasizing that security must be designed into each component rather than added as an afterthought.

This review adopts a structured search strategy inspired by systematic review practices. All studies were retrieved from IEEE Xplore for the years 2020–2024, using search strings combining IoT security, machine learning, deep learning, and intrusion detection. Papers were included if they (i) addressed IoT-specific security issues, (ii) applied ML/DL as core software components, and (iii) reported empirical findings. Non-peer-reviewed and purely conceptual works were excluded. In total, 12 studies met these criteria and were categorized into two groups; ML-based and DL-based. Each was analyzed not only for detection accuracy but also for software-relevant aspects such as deployment feasibility, integration complexity, computational overhead, and software dependencies.

The solution to these challenges is the emergence of smart security mechanisms using cutting-edge Artificial Intelligence (AI)-based technologies. AI-based techniques have been playing a critical role in improving the security of IoT [3]. They lead to adaptive security frameworks that can detect and respond to IoT threats in real-time. AI-powered solutions can proactively counter advanced cyber-attacks, by detecting patterns and anomalies in IoT networks, thereby preserving the integrity and resilience of IoT systems [4]. IoT devices can also adapt and learn from past threats using traditional and advanced ML algorithms by appropriately perceiving unusual activity in the future hence enhancing security with less reliance on human intervention [5, 6].

Traditional security models are not sufficient to protect IoT systems because they often rely on hard-coded security policies and signature-based detection engines that match threats against predefined rules [7]. While effective for known attacks, these static mechanisms fail to adapt to new or evolving threats. Moreover, variations in software implementations across heterogeneous IoT platforms, combined with the scale of deployment, create scalability challenges that prevent uniform enforcement of security protocols. These limitations highlight the need for adaptive, software-driven security solutions based on modern AI techniques for IoT systems [8].

Machine Learning (ML) and Deep Learning (DL) have emerged as crucial techniques in improving the security of IoT networks. These techniques are highly effective for processing large volumes of data to find patterns that may be indicative of potential threats. ML and DL can be used to identify anomalies and malicious activities in real-time using sophisticated algorithms that allow quick responses to breaches [9]. As an IoT device provider, manufacturers need to be proactive with the data by identifying, addressing, and managing risks associated with these devices as attackers are always on the lookout for potential vulnerabilities and ways to exploit them. The application of ML and DL models can inherently enable ongoing learning with an evolving ability to cope with new data patterns, providing an improved ability to identify future automatic malicious attacks [10]. The potential benefits achieved by implementing ML DL approach for IoT security are presented in Figure 2.

Unlike prior surveys that mainly catalog ML/DL

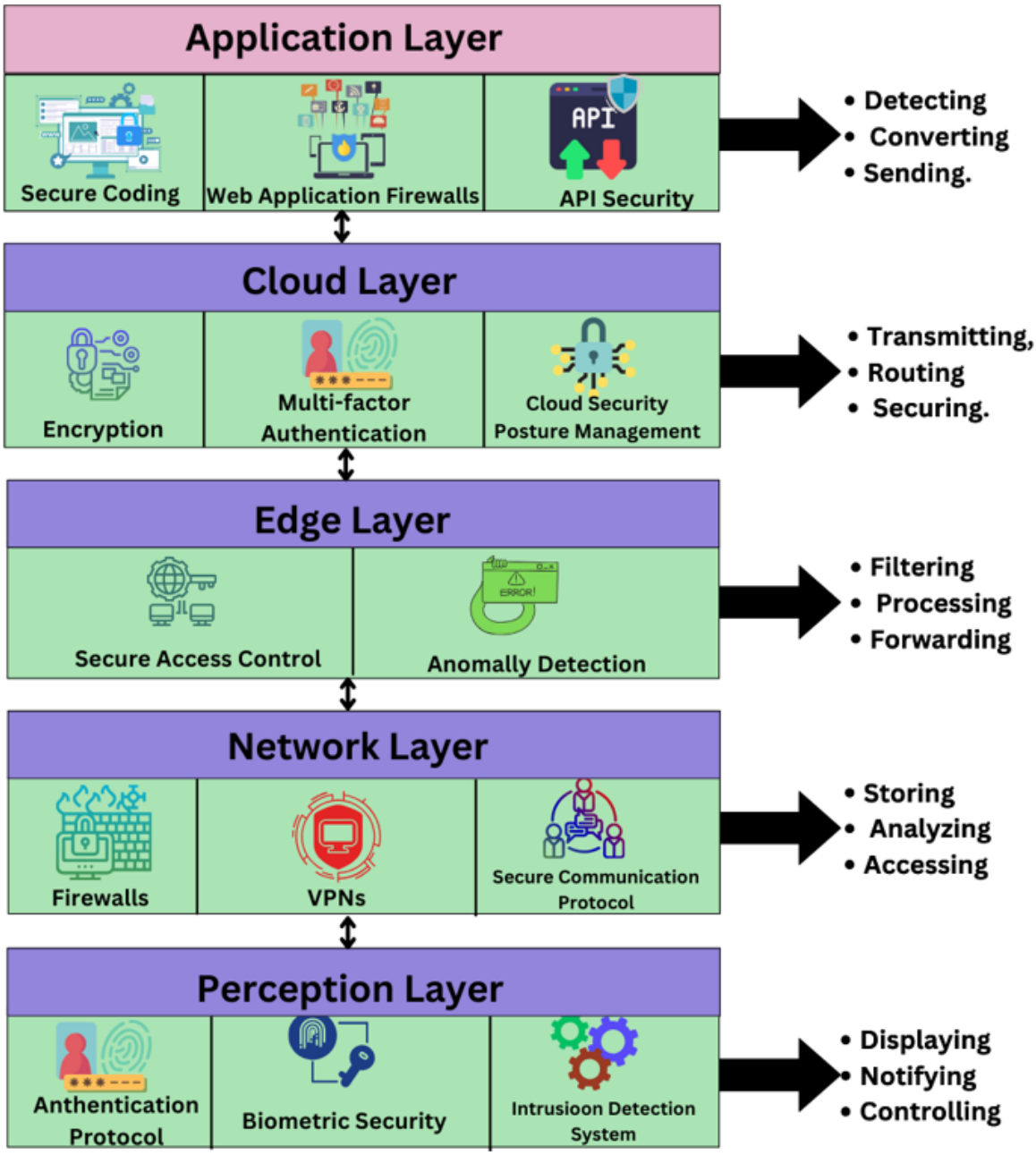


Figure 1. IoT security framework.

techniques and benchmark their detection accuracy, this review contributes a software engineering–centric synthesis. Specifically, it evaluates security models in terms of deployment feasibility (cloud, edge, or device), integration complexity (library, service, or firmware module), computational overhead (RAM/CPU limits), and software dependencies (framework requirements). By reframing challenges such as adversarial robustness, scalability, and lifecycle drift as software testing, architectural, and maintainability problems, this review offers insights that existing surveys have largely overlooked.

This review is positioned within the field of software engineering along with cybersecurity perspective. Our central interest is how ML/DL-based security solutions integrate into IoT software systems. To this end, the review is guided by the following research questions:

- **RQ1:** How are ML/DL-based IoT security solutions designed and deployed across different software environments (cloud, edge, device)?
- **RQ2:** What computational and dependency constraints (RAM, CPU, frameworks) limit deployment on resource-constrained IoT devices?
- **RQ3:** How do software lifecycle concerns—such as retraining, OTA updates,

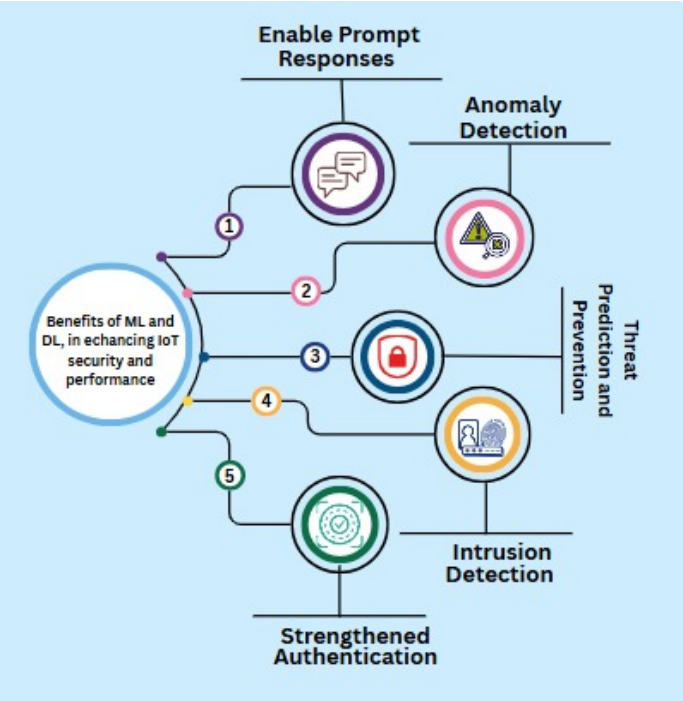


Figure 2. Enhancing IoT security with machine and deep learning approaches.

and maintainability—affect the long-term reliability of these solutions?

The contributions of this paper are as follows:

1. Software engineering lens: We review recent ML- and DL-based IoT security studies (2020–2024) not only for detection accuracy but also for software-relevant aspects such as deployment models, integration complexity, computational overhead, and framework dependencies.

2. Comparative synthesis: We analyze trade-offs between lightweight ML techniques that can run on constrained devices and resource-intensive DL approaches that are more practical in edge or cloud environments.

3. Practical guidance: We outline implications for software developers and architects, emphasizing lifecycle considerations such as development, integration, monitoring, and updating of ML/DL-based IoT security modules.

The rest of this paper is organized as follows: recent case studies of security attacks in IoT environments are discussed in Section 2. Section 3 presents ML-based security solutions; then it examines the ML techniques adopted for intrusion detection in IoT networks. DL-driven approaches are described in section 4 while discussing advanced models to discover cyber threats in real-time. Section 5 presents the results obtained from the review and discusses their implications. Section 6 describes the challenges in terms of IoT security and future research directions. Lastly, in section 7, the paper is concluded with a summary of its findings and a call for systems integrating an AI-driven security framework.

2 Recent attacks on IoT devices

IoT devices are the primary targets for cybercriminals owing to the dynamic growth of IoT technology. Over the past few years; the retail market, manufacturing, and healthcare IoT systems have been badly affected based on the crucial vulnerabilities causing data breaches, service disruptions, and physical harm. Real-world case studies provide crucial information regarding dynamic cybersecurity attacks. We can analyze the effectiveness of current security mechanisms by determining common vulnerabilities and frequent attack patterns. Below section will discuss recent attacks on IoT devices causing damage to industrial sectors, healthcare devices, and smart home devices. These case studies were selected based on the diversity of attack techniques and their significance in the current security landscape. The potential damages of IoT-based systems are graphically presented in Figure 3.

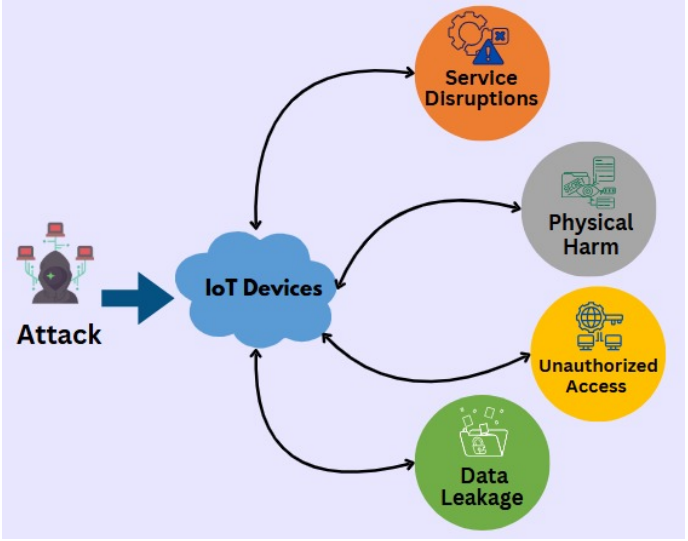


Figure 3. Security vulnerabilities in IoT devices.

2.1 Incident: Security Breach in PSA-Certified IoT Chip

A critical security issue was found in an IoT security chip that was certified against the Arm Platform Security Architecture (PSA) Level 2 standard [11]. The chip was extensively embedded in smart home

devices, industrial IoT systems, and authentication systems, where it used AES-128 to protect sensitive data. However, researchers discovered that the certified chip remained susceptible to a non-invasive electromagnetic (EM) side-channel attack, enabling attackers to obtain parts of the encryption key by passively observing the electromagnetic signals leaked during encryption processing. This vulnerability casts serious doubt on the ability of certification standards to protect IoT hardware against advanced physical attacks.

The intruders used an electromagnetic probe and an oscilloscope to chart emissions from the chip as it conducted encryption work. With thousands of EM traces, the attackers drove secret patterns in how the chip processed encryption keys. Statistical T-tests and correlation analysis verified a significant data leak that permitted attackers to recover almost 50% of the 16-byte AES encryption key. Until half of the key is revealed it is impossible to use brute-force techniques to recover the remaining bytes, which would greatly speed things up in terms of time and computational power required to break the encryption. The execution of this attack without direct physical tampering of the chip demonstrated the stealthy and dangerous nature of side-channel attacks in IoT security.

These findings led many security experts to call for stronger countermeasures to protect IoT devices from this kind of vulnerability. The recommendation was to upgrade to PSA Level 3 certification, which requires stronger safeguards against side-channel attacks. Instead, experts proposed hardware-level changes, like masking techniques and noise injection, to block EM signal leaks. The researchers also proposed software-level security improvements, including secure key management and advanced cryptographic techniques, to reduce the risk of similar attacks in future IoT deployments, in addition to hardware enhancements. One of the key learnings from this case study is the need to strengthen IoT security across both hardware and software layers so that critical devices can withstand an evolving threat landscape.

From a software engineering perspective, this incident highlights the importance of robust firmware update pipelines and secure cryptographic libraries. Without well-engineered software patching mechanisms, even certified hardware remains vulnerable, emphasizing that IoT security is as much a software lifecycle problem as it is a hardware design issue.

2.2 Incident 2: Baby Monitoring Camera Hijacking

An alarming security incident involved baby monitoring cameras (BMCs) which reported that attackers obtained unauthorized access to live video streams and interacted with children [12]. The breach was linked to several security vulnerabilities, including default login credentials, unencrypted peer-to-peer (P2P) cloud streaming, and open TCP ports (554, 5000). These vulnerabilities enabled hackers to hack the cameras remotely, watch live feeds, and change the camera's settings — a serious privacy threat to families. In some instances, hackers raided home networks by scanning for vulnerable cameras and then brute-forcing default admin credentials. Once they were inside, they exploited intercepted cloud-based communications to control the device remotely, often speaking through the camera's built-in microphone to scare or manipulate the users. Parents whose kids had been affected reported unsettling incidents where strangers addressed their children at night or shouted orders using hooked-up smart home systems.

To mitigate these threats, manufacturers introduced required password changes at the time of setup, improved encryption for data sent over the cloud, and reduced the number of unnecessary open ports. Users were also urged to turn off remote access features when they are not needed to use strong, unique passwords and to keep firmware up to date to head off similar breaches in the future.

From a software engineering standpoint, the vulnerabilities reveal poor default software configurations and weak authentication modules in IoT firmware. Addressing these issues requires secure-by-design development practices, where access control and encryption are embedded into the software stack rather than retrofitted after deployment.

2.3 Incident 3: Voice Assistant Exploitation

A major security breach was discovered in smart home voice assistants where attackers exploited third-party applications (skills/actions) [12]. The vulnerability was an extensive security breach in smart home voice assistants, including Amazon Echo and Google Assistant, which were targeted by attackers exploiting third-party applications (skills/actions) to bypass security checks and eavesdrop on user conversations. The researchers said malicious applications could be programmed so that they would not stop running after sending a command, allowing it to surreptitiously

listen in on private conversations without the user being aware. They also used silent pauses and unpronounceable characters to deceive users into disclosing sensitive information like passwords and financial data.

The attackers exploited loopholes in the voice assistant's skill verification process to gain excessive access to malicious applications on user devices. After a user had unwittingly installed the compromised skill, attackers could eavesdrop on conversations in real time, or conduct phishing attacks by impersonating system messages and asking users to disclose their credentials. Smart assistants are present in several IoT ecosystems. The breach in question represented a major privacy and security risk to both smart homes and enterprise environments.

In response to this threat, security professionals advised implementing more rigid verification processes, persistent monitoring of voice assistant applications, and improved permission management for third-party apps. Users were also advised to periodically check installed skills, disable any applications not in regular use, and avoid sensitive discussions near smart assistants. The companies behind these devices tightened their security audits of third-party applications; ensuring voice-based interactions cannot be manipulated to trigger cyber-attacks.

For software engineers, this incident underscores the risks of insufficient software validation pipelines for third-party applications [13]. Stronger software auditing frameworks, sandboxing mechanisms, and runtime monitoring modules are necessary to prevent malicious skill integration into IoT platforms.

2.4 Incident 4: Exploiting Medical IoT Devices for Cyberattacks

Hackers attacked hospital-based IoT-enabled equipment, such as MRI machines, infusion pumps, and patient monitoring systems, by exploiting weak authentication mechanisms and default credentials left unchanged [14]. These devices were infected and became part of a botnet, enabling attackers to swamp targeted networks with huge volumes of traffic, disrupting critical healthcare services. Security experts discovered a common vulnerability on used hospital IoT networks, where hacked medical devices were being used to perform DDoS attacks.

In this incident, "MedJack," a botnet malware was found inside hospital networks, slowly infecting at-risk medical IoT devices. Unlike conventional malware, which was directed at consumer and enterprise systems that featured better monitoring to discover intruders, the MedJack was designed to go undetected, hiding inside medical equipment that didn't have such monitoring. The attackers exploited these infected devices to carry out DDoS attacks against hospital databases and medical record systems, resulting in network slowdowns and disruptions in patient care Emergency treatments were delayed in some cases, when hospitals lost access to critical patient records.

To defend against these risks, hospitals were recommended to segment their networks so that medical IoT devices could operate over isolated, secure networks that weren't intertwined with administrative or patient data systems. Also helped were intrusion detection systems (IDS) that helped detect unusual traffic patterns early on before larger disruptions could take place. The best practices to keep hospital IoT environments more secure against similar cyber threats included regular software updates, enforcing strong passwords, and disabling unnecessary remote access.

This case illustrates a gap in software maintainability: many medical IoT devices lacked structured update mechanisms or modular software designs that allow timely patching. For safety-critical domains, software engineering practices such as continuous integration/continuous deployment (CI/CD) for firmware updates and formal verification of security modules become essential.

2.5 Incident 5: Aircraft Avionics Vulnerability

A critical security vulnerability had been found in Boeing 737 and 787 aircraft avionics systems where unprotected servers opened up unauthorized access to sensitive firmware updates [12]. Researchers discovered that avionics-related software had been made publicly available, allowing hackers to download, analyze, and exploit vulnerabilities in core flight systems. Among the vulnerabilities were buffer overflow exploits, insecure file transfer protocols, and the potential for remote code execution.

Reverse-engineering the avionics software, researchers showed that a plane's control systems could be hijacked so that malicious firmware updates could be injected. In turn, it could result in spurious sensor readings, interruptions of the autopilot, or the ability to remotely take control of some functions in an aircraft. Furthermore, had adversaries manipulated the data as

it was transmitted to a flight controller, the risk would have been amplified since there were no mechanisms in place to authenticate or encrypt this data.

After the detection, airline manufacturers and regulators implemented secure firmware update policies, encryption of avionics information, and limited server access to zero in on one-off downloads. Multi-factor authentication and network segmentation were also used to ensure aircraft systems only received validated updates. These incidents highlight the pressing need for integrated, robust IoT security frameworks that span hardware, software, AI-driven anomaly detection, and continuous monitoring. As IoT adoption continues to rise, IoT security should also become a priority to protect systems against constantly evolving cyber threats and ensure the safety, privacy, and reliability of interconnected systems.

From a software lifecycle perspective, the vulnerability highlights weaknesses in software distribution pipelines. Secure build environments, cryptographic signing of firmware, and automated integrity checks should be integral to avionics software engineering processes to prevent tampering during updates.

A comparative table summarizing the incidents based on attack type, vulnerability, exploited protocol, and potential ML/DL countermeasure is presented in Table 1.

3 Machine learning – driven approaches for Safeguarding IoT Networks

Traditional security mechanisms have proven slow to respond to the increasing complexity and volume of cyber threats. ML has become a powerful instrument for strengthening IoT security, providing adaptive, real-time capabilities for threat detection and mitigation. In the next section, we will describe different ML-based techniques that improve the security of IoT devices against complex attacks and maintain low false positives and system overhead [15, 16]. Researchers have explored all three ML-based approaches including supervised, unsupervised, and hybrid techniques to ensure the security mechanisms in IoT systems. A step-by-step process to apply an ML-based approach for IoT security is presented in the flowchart in Figure 4.

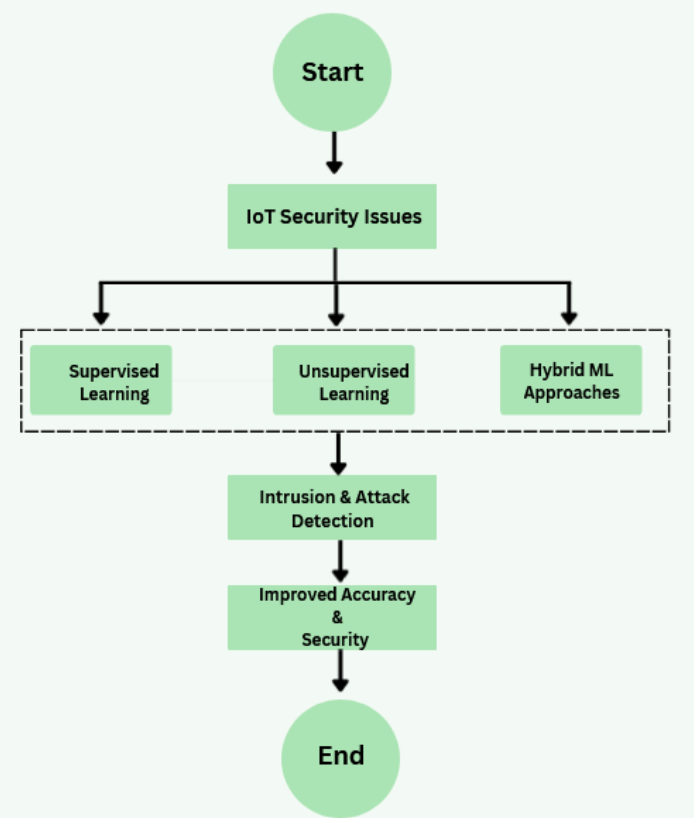


Figure 4. Flowchart of ML approach for IoT security.

Digital forensics researchers introduced a novel IPS

Table 1. Summary of IoT attacks, vulnerabilities, and ML/DL countermeasures.

Incident	Attack Type	Vulnerability	Exploited Vector	ML/DL Countermeasure
1. PSA IoT Chip	EM side-channel	Weak AES-128, hardware leak	EM emissions	Anomaly detection on EM patterns
2. Baby Monitor Hijack	Remote access	Default creds, open ports	TCP/IP, cloud streaming	DL-based intrusion detection
3. Voice Assistant Exploit	Eavesdropping	Weak app verification	Third-party apps	Behavioral anomaly detection
4. Medical IoT (MedJack)	Botnet/DDoS	Default creds, weak auth	Hospital IoT network	ML network anomaly detection
5. Aircraft Avionics	Firmware hijack	Unprotected updates	Firmware update servers	ML integrity verification

using ML techniques for Industrial IoT (IIoT) and Cyber-Physical Systems (CPS) [17]. They introduced a novel family of classifiers, termed z-classifiers, that are tailored to achieve zero false positive detection of malicious activities, in contrast to classic classifiers that focus solely on accuracy. They described an iterative learning firewall that adapts itself to the existing labeled Intrusion Detection System (IDS) data and creates a rule-based security model for preventing the false positive access of legitimate traffic. The approach was evaluated on both a power grid monitoring system and the KDD CUP'99 dataset, which shows the effectiveness of the proposed method for reducing false positives while keeping an acceptable false positive rate. The research demonstrated the feasibility of adaptive security models powered by ML for IoT systems or environments where it is crucial to provide continuous operation.

From a deployment perspective, this solution is primarily designed for edge-based deployment, since continuous operation with minimal false positives requires local processing close to the devices. For software architects, this highlights the challenge of integrating lightweight ML classifiers into resource-constrained firmware while keeping the IDS adaptable. Integration-wise, the model is closest to a firmware-level library that can be embedded into existing IDS modules. However, its custom classifier design may require substantial re-engineering of legacy IoT software stacks, increasing adoption complexity.

In a separate study [18], researchers designed an ML-based framework for spam detection in the IoT environment. The authors computed a spamicity score for IoT devices using five different ML models; Bayesian Generalized Linear Model (BGLM), Boosted Linear Model, eXtreme Gradient Boosting (XGBoost), Generalized Linear Model (GLM) with Stepwise Feature Selection, and Bagged Model. They applied feature engineering methods like Principal Component Analysis (PCA) and entropy-based filters to improve feature selection for spam detection. The REFIT Smart Home dataset was used to validate the proposed system, reporting a significantly higher accuracy in detecting malicious interactions than previously applied techniques.

The framework functions as a cloud-side service, making it naturally suited for integration as a standalone microservice. Yet, it lacks clear APIs or modular interfaces, meaning software engineers would need to wrap the models manually to connect with IoT middleware.

From a software engineering perspective, however, this framework assumes continuous access to high-quality labeled data and does not discuss how model updates or retraining would be incorporated into IoT software pipelines. Its deployment feasibility on constrained edge devices also remains unclear, highlighting the need for lightweight implementations and automated update mechanisms to ensure maintainability in real-world systems.

In a similar work, researchers proposed an ML-based method for attacking repetition in IoT networks [19]. The authors used an advanced ML algorithm XGBoost, which is a gradient-boosted decision tree technique that can analyze network traffic, to detect anomalous behavior consistent with the behavior of compromised IoT devices. Using the IoT – dataset, researchers trained and evaluated a model of benign and malware IoT traffic. In their experiments, they achieved 93.6% accuracy with a high recall of 99.9% to ensure that compromised devices are detected while avoiding false negatives. This work not only emphasized the effectiveness of Xgboost concerning traditional classification methods but also demonstrated that information is vital for strengthening the security of the IoT systems, namely, in detecting unauthorized device activity in real-time.

Given its lightweight design, the model could be deployed as a firmware-embedded library within IoT device software. While this supports low-latency detection, it complicates integration with existing IoT operating systems, as firmware updates would be required each time the model is retrained.

Furthermore, it was also previously examined from the perspective of adversarial ML to evade traditional ML-based malware detection within an IoT setting [20, 21]. The working principle of evasion attack techniques proposed by the authors are as follows: A feature similarity attack in which the authors use Euclidean Distance (ED) to re-cast information in different populations to generate adversarial malware samples similar to benign applications using an optimization algorithm. Researchers targeted genuine Android apps from real-world datasets i.e., AndroZoo and AMD. They succeeded in completely identifying threat risk from PSO with 89.6% accuracy using ED classifiers, outperforming traditional ML-based classifiers (SVM, RF, LR). As adversarial attacks continue to pose growing

challenges for ML applications, this is one area where feature-space manipulations can easily deceive even state-of-the-art ML-based malware detectors. This can lead to the design of more resilient defense mechanisms to resist adversarial attacks. The results highlight the shortcomings of static feature-based ML models for malware detection and suggest incorporating adversarial training to enhance IoT security.

The proposed defenses are primarily conceptual and not packaged as software modules. In practice, they would require integration as additional layers within malware detection engines, increasing system complexity and potentially demanding modifications to API-level communication.

In a separate study [22], researchers applied Random Forest (RF), one of the ML approaches to enhance the security recommended solution for the 5G networks. RF classifiers were employed to identify malware, DDoS attacks, and attempts to breach networks based on network traffic data from a large number of users. The authors compared RF with other ML techniques including SVM and DT, establishing considerable improvement in the accuracy and efficiency of classifying secure and insecure traffic. The proposed models have been trained using real-time 5 G-based network dataset and have shown to be capable of achieving a detection rate while reducing false positives. According to the study, RF-based models represent a scalable and effective solution for securing the 5G network, demonstrating their potential to enhance IDS for next-generation wireless communication scenarios.

This solution would integrate best as a cloud-side microservice connected through standardized APIs. However, the paper does not address interoperability with existing IoT middleware, leaving architects to resolve compatibility with heterogeneous vendor systems.

In another study, researchers recently proposed a framework driven by ML to secure IoT-based data transmission [23]. They examined the limitations of conventional cryptographic approaches in the context of IoT devices and applied anomaly detection, intrusion detection, and encryption using ML to secure sensitive information. They implemented Isolation Forest (iForest) to detect anomalies, and SVM as Intrusion detection while also encrypting the data using Advanced Encryption Standard (AES). Experimental results on practical IoT datasets achieved a high accuracy of 99.5% for anomaly detection, 98.61% for intrusion detection, and fast speed encryption as well as low overhead performance. This research demonstrated that ML-based security mechanisms outperform traditional rule-based cryptographic methods, thus they can be utilized for IoT networks' secure design.

The framework is modular and aligns well with a microservices architecture: anomaly detection and IDS components can operate as independent services, while encryption can be embedded at the middleware level. This modularity makes integration feasible, but dependency management across services could complicate lifecycle maintenance.

In addition, many researchers have examined the use of different ML-based methods to identify and prevent DoS and DDoS attacks for IoT networks [24]. The study evaluated several ML models, like SVM, ANN, KNN, DT, and RF for detecting malicious and normal traffic. They analyzed both centralized and distributed deployment methodologies in the paper while highlighting the effectiveness of distributed techniques in the identification of early attacks. By leveraging real-world datasets (CICIDS2017, IoTPOT) and validating that hybrid ML approaches i.e., K-Means along with DT, research led to lower false positive rates. The researchers conclusively proved the superiority of these methods over individual models. Experiments revealed that anomaly-based ML detection methods provide superior DDoS mitigation effectiveness over signature-based techniques in general, suggesting the former can adapt to new attack patterns much better than the latter. The summary of ML techniques applied for IoT security threats is graphically presented in Figure 5.

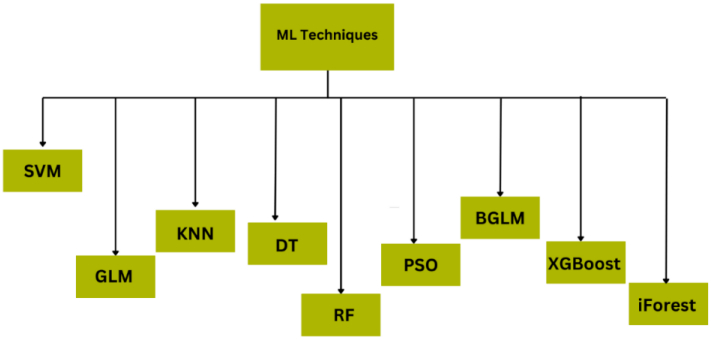


Figure 5. ML techniques applied for IoT security.

3.1 Summary

Recent research applied various ML techniques including XGBoost, RF, SVM, and iForest for IoT intrusion detection, anomaly detection, and cyber threat mitigation showing their efficiency in this study. With adaptive learning, automated threat classification, and reduced false positives, these approaches produce improved results as compared to the traditional rule-based security mechanisms. Nonetheless, challenges pose serious concerns such as adversarial attacks, limited datasets, and testing ML-based solutions in real time on resource-constrained IoT devices.

Addressing adversarial attacks is crucial to strengthen ML-based models for robust IoT security measures. Adversarial training is the most widely followed technique in which model training is performed both on cleaned data and thoughtfully designed adversarial examples which expand the flexibility against modified input data. Techniques such as defensive distillation and input preprocessing methods improve the model's response against minor changes in input. Moreover, feature squeezing, randomization, and data sanitization methods can neutralize adversarial noise prior to data feeding into the models.

4 Strengthening IoT Networks Against Cyber Threats Using Deep Learning

Traditional ML-based security mechanisms have limitations against modern techniques being used in real-world attacks, as IoT networks have become increasing targets of cyber threats. Deep learning (DL) is an advanced approach in AI that has growing applications in many areas [25]. DL models process large amounts of network traffic data, capturing subtle attack patterns that traditional rule-based and ML methods may overlook. DL techniques, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Autoencoders, and Generative Adversarial Networks (GANs), facilitate real-time intrusion detection and allow adaptive cybersecurity measures [26]. In this section, we highlight the most common deep learning methods applied to protect IoT networks and their efficacy in identifying cyber-attacks.

Researchers addressed the IoT security problem of vulnerability identification and penetration testing to detect zero-day vulnerabilities within smart environments [27]. The authors proposed Long Short-Term Memory – Recurrent Neural Network-enabled Vulnerability Identification (LSTM-EVI), a deep learning-based penetration testing framework to identify scanning attacks and security weaknesses of IoT devices. For testing the model, they built a smart airport cybersecurity testbed that integrated physical IoT devices with simulated environments to replicate the realistic network conditions. The results indicated that the LSTM-EVI system reached 99% detection accuracy among these scanning attacks, remarkably outperforming the performance of the other ML models such as MLP, SVM, NB, and KNN. They observed that LSTM-based deep learning models are highly efficient in boosting the IoT vulnerability detection process, thus enabling a proactive application of security measures in smart surroundings by anticipating cyber threats.

This framework resembles a standalone penetration-testing application rather than a library or micro service. For real-world adoption, it would need to be refactored into modular services or integrated APIs so software teams can embed it into existing IoT security pipelines [28]. The LSTM-EVI model incurs moderate to high memory and CPU overhead due to its sequential processing requirements, making it unsuitable for direct deployment on lightweight IoT firmware. Software engineers would need edge gateways or cloud servers with GPU support to run such penetration-testing models effectively.

In another study [29], authors proposed a solution addressing IoT security threats through an intrusion detection mechanism, observing that a real-time detection mechanism of cyber threats could be developed by identifying gateway traffic. A hybrid DL-based IDS was proposed that employed CNNs to extract spatial features along with LSTM networks to analyze temporal features. They used the CSE-CIC-IDS2018 dataset which is a comprehensive and current intrusion detection dataset with multiple attack scenarios for training and evaluating the model.

The experimental results showed that the CNN-LSTM model reached 99% accuracy for binary classification and 97.11% for multiclass classification, exceeding prior DL-based IDS approaches. However, such results were obtained on a curated dataset under controlled conditions. In real-world IoT environments, software noise, heterogeneous firmware, and evolving traffic patterns may significantly reduce reliability. Moreover, the study does not address how model drift would be managed in practice, particularly when

IoT nodes receive over-the-air (OTA) updates that alter their behavior. Without lifecycle mechanisms for retraining and redeployment, sustaining this accuracy in operational systems remains uncertain.

The hybrid CNN-LSTM model has high computational cost, as CNN layers demand significant RAM for feature maps while LSTMs require sequential memory operations. This overhead restricts its deployment to edge or cloud nodes, and software engineers must plan for GPU acceleration or hardware optimization (e.g., quantization, pruning) to reduce latency. The CNN-LSTM model is best suited as a cloud or edge microservice that can be queried by IoT middleware. Direct embedding into device firmware is impractical due to computational cost, so integration requires containerized deployment and well-defined APIs for interoperability.

The work done in [30], focused on autonomous intrusion detection and cyber-physical system security using Deep Reinforcement Learning (DRL). The authors proposed multiple approaches based on DRL such as multi-agent DRL models for cyber defense, IDSs, and game-theoretic simulations to mitigate the growing cyber threat landscape. Researchers analyzed different datasets to evaluate DRL performance concerning: 1) network intrusion datasets, 2) real-world cyber-physical system logs, and 3) adversarial cybersecurity simulations. The findings showed that models utilizing DRL outperformed conventional security methodologies, offering dynamic, scalable threat identification and mitigation. The results indicated that IoT networks can be more robust and equipped to cope with highly advanced and rapidly new cyber threats using DRL as a cybersecurity solution.

DRL approaches are among the most resource-intensive, requiring extensive CPU/GPU cycles for training and inference. While powerful in simulations, deploying DRL modules in IoT systems raises severe feasibility challenges, as real-time adaptation cannot be achieved on-device without specialized accelerators. DRL-based solutions align more with orchestrated micro services or simulation platforms rather than embedded libraries. Their complexity makes them suitable for centralized or federated architectures, where software engineers must design adaptive interfaces between the DRL engine and IoT monitoring services.

The study [31] addressed the cyberattack detection problem in IoT networks, with a focus on data scarcity and heterogeneous feature spaces in intrusion detection systems. The researchers proposed a framework called Federated Transfer Learning (FTL), a novel approach by combining the capabilities of Federated Learning (FL) and Transfer Learning (TL) to improve the performance of DL models at identifying cyberattacks. For the evaluation of the proposed framework, four real-world cybersecurity datasets were used, namely N-BaIOT, KDD, NSL-KDD, and UNSW-NB15, having different kinds of IoT-related security threats like DDOS, botnet attacks, and network intrusion. Experimental results proved the efficiency of the FTL framework since it reached 99% accuracy and improved the accuracy by 40% compared with conventional unsupervised deep learning methods that were used in diverse IoT systems. This highlights the potential of collaborative DL models with a focus on acquiring performance measures in securing IoT networks, especially under conditions of limited labeled data and heterogeneous network environments.

FTL distributes some workload to edge devices, but even with partial training offloaded, deep transfer learning consumes substantial memory and CPU. Engineers must balance model accuracy against the practical energy and latency constraints of IoT nodes, potentially requiring lightweight surrogates or knowledge distillation. FTL frameworks naturally integrate as distributed micro services, with federated clients running on edge devices and a coordinating server in the cloud. For architects, this requires careful design of software APIs and secures communication protocols to manage updates across heterogeneous IoT systems [32].

A recent research targeted intrusion detection tasks in IoT networks, aiming the secure resource-constrained IoT nodes against diverse cyber threats [33]. They presented Deep-IDS, an LSTM-based Intrusion Detection System (IDS) for the task of real-time intrusion detection and mitigation. The model was trained using the CIC-IDS2017 dataset that included diverse attacks such as DOS, DDoS, Brute Force, Man-in-the-Middle (MITM), and Replay Attacks. The experimental results showed that Deep-IDS can accurately classify benign (normal) and malicious (attack) traffic, with an accuracy of 97.67%, a detection rate of 96.8%, and a low false alarm rate. The results emphasized that Deep-IDS is ideal for edge-server implementation, offering low-latency, high-accuracy threat detection to protect IoT networks against constantly evolving cyber threats.

Table 2. Summary of ML and DL approaches to mitigate IoT security challenges.

Study	Year of publication	Proposed Approach	ML Techniques Used	Dataset Used	Key Findings
[17]	2020	z-classifiers to achieve zero false positives in detecting malicious activities	Custom z-classifiers, iterative learning firewall	KDD CUP'99, Power Grid Monitoring System	Reduced false positives while maintaining a reasonable false negative rate
[18]	2020	Spamicity score development using ML-based classification for IoT spam detection	BGLM, Boosted LM, XGBoost, GLM, Bagged Model	REFIT Smart Home Dataset	Improved accuracy in identifying malicious IoT activity, reducing false positives
[24]	2021	ML techniques for DDoS detection and mitigation	SVM, ANN, KNN, Decision Trees, Random Forest	CICIDS2017, IoTPOT	Hybrid ML models (K-Means + Decision Trees) reduced false positives and improved attack detection
[27]	2021	Vulnerability identification and zero-day attack detection	LSTM-EVI – An LSTM-based penetration testing framework	Smart Airport Cybersecurity Testbed (Physical IoT + Virtual Simulation)	Achieved 99% detection accuracy, outperforming MLP, SVM, Naive Bayes, and KNN
[19]	2022	Anomaly detection and replication attack identification	XGBoost (Gradient-Boosted Decision Trees)	IoT-23 Dataset	Achieved 93.6% accuracy and 99.9% recall, proving superior to traditional models
[20]	2022	Explored adversarial ML attacks to bypass IoT malware detectors	Euclidean Distance (ED), Particle Swarm Optimization (PSO)	AndroZoo, AMD datasets	100% evasion success (PSO), 89.6% evasion success (ED), highlighting weaknesses in ML malware detection
[22]	2023	Detect malware, DDoS attacks, and intrusions in 5G networks	RF, SVM, Decision Trees	Real-World 5G Network Dataset	RF demonstrated higher accuracy and efficiency in intrusion detection
[30]	2023	Autonomous intrusion detection and cyber-physical system security	DRL for multi-agent cyber defense and intrusion detection	Network intrusion datasets, CPS logs, adversarial cybersecurity simulations	DRL models outperformed traditional security mechanisms, enabling adaptive real-time threat detection
[23]	2024	Anomaly detection, intrusion detection, and encryption for IoT security	Isolation Forest (Anomaly Detection), SVM (Intrusion Detection), AES (Encryption)	Real-World IoT Dataset	Achieved 99.5% anomaly detection accuracy, 98.6% intrusion detection accuracy, with minimal processing overhead
[29]	2024	Intrusion detection in IoT networks for real-time cyber threat analysis	Hybrid CNN-LSTM Intrusion Detection System (IDS)	CSE-CIC-IDS2018	Achieved 99% accuracy (binary classification) and 97.11% accuracy (multiclass classification), outperforming existing IDS models
[33]	2024	Cyberattack detection in IoT with limited labeled data and heterogeneous networks	FTL integrating FL and TL for collaborative deep learning-based intrusion detection	N-BaIOT, KDD, NSL-KDD, UNSW-NB15	Achieved 99% accuracy, improving by 40% over unsupervised DL approaches, proving efficiency in diverse IoT environments
[34]	2024	Real-time anomaly detection and classification of malicious IoT network traffic	Hybrid CNN-LSTM IDS for spatial and temporal pattern recognition	CICIOT2023, CICIDS2017	Achieved 98.42% accuracy, F1-score of 98.57%, and low loss rate of 0.0275, outperforming traditional IDS models

Although Deep-IDS demonstrated strong detection accuracy, its LSTM-based structure consumes more RAM than typical IoT nodes can provide. It is best suited for edge servers with more memory, where engineers can manage retraining and updates without exhausting device resources. Deep-IDS can be packaged as an edge-side microservice or library linked to local IoT gateways. However, frequent model retraining may complicate lifecycle management, requiring automated deployment pipelines and rollback strategies within IoT software stacks.

The aim of the study [34] was to address the intrusion detection problem in IoT networks while particularly focusing on the difficulties related to detecting real-time threats and classifying malicious network traffic. The authors suggest a hybrid deep learning model approach for an IDS, employing CNNs for spatial feature extraction and LSTM networks for temporal pattern recognition. The CICIoT2023 dataset is used to train and evaluate the model involving different types of IoT attacks including DDoS, brute force, spoofing, and web-based attacks, while the testing was performed on the CICIDS2017 dataset. The experimental results showed that 98.42% accuracy is achieved by the CNN-LSTM model with a 0.0275 low loss rate and an F1-score is 98.57% proving their system is far better than the existing techniques of intrusion detection. The results indicated that deep learning-based IDS models present a promising method for effective real-time anomaly detection, reinforcing IoT network security in the face of emerging cyber threats. DL technique used for the security of IoT is shown in graphical form in Figure 6.

The CNN-LSTM IDS imposes heavy computational overhead, with CNN layers requiring high parallelism and LSTMs adding sequential delays. For resource-constrained IoT devices, such models must be offloaded to edge/cloud services or optimized via compression techniques like quantization and model pruning. This CNN-LSTM IDS is positioned as a standalone application for IoT network monitoring. To integrate with existing IoT platforms, it would need to be modularized into containerized services or exposed as an API, since embedding the full DL stack into device firmware is impractical.

4.1 Summary

While existing studies report strong detection accuracy, their software engineering implications remain underexplored. Most solutions assume cloud deployment, leaving open questions about latency and bandwidth, while only a few consider lightweight edge or on-device use. Integration is also a challenge, as models are often presented as prototypes rather than modular libraries or microservices. Computational demands further limit adoption: ML models like RF and XGBoost can run on constrained devices, but CNNs, LSTMs, and DRL architectures typically require powerful edge or cloud resources. Finally, software dependencies are seldom discussed; reliance on full DL stacks such as TensorFlow or PyTorch complicates IoT deployment unless ported to lightweight runtimes like TensorFlow Lite or ONNX. These gaps suggest that future work must evaluate IoT security solutions not only for accuracy but also for deployment feasibility, integration complexity, resource use, and dependency management. The summary of ML and DL approaches to mitigate IoT security challenges is presented in Table 2.

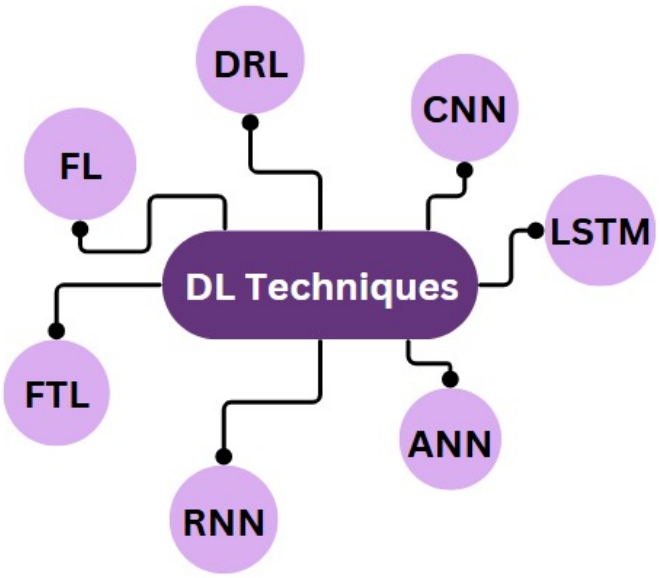


Figure 6. DL application for IOT security.

A year-wise distribution of selected studies is presented in Figure 7.

A Summary of ML/DL-based IoT security approaches with software-relevant considerations (inference overhead, model size, deployment feasibility) is presented in Table 3.

5 Results and Discussion

The findings from the reviewed studies highlight both opportunities and limitations in applying ML/DL for IoT security. A recurring trade-off emerges between accuracy and feasibility: while deep models such as CNN-LSTM and DRL often report accuracies above 97%, their high memory and CPU demands restrict

Table 3. Comparison of ML- and DL-based IoT security approaches with software-relevant considerations.

Study	Year	Technique	Reported Accuracy	Inference/Overhead	Approx. Model Size	Preferred Deployment
[17]	2020	Custom ML Classifier	Zero false positives (IDS)	Lightweight	Small (few MB)	Edge/Device
[18]	2020	Multiple ML models (XGBoost, GLM, etc.)	High accuracy	Moderate (requires feature engineering)	Medium	Cloud
[19]	2022	Gradient-boosted trees	93.6% accuracy, 99.9% recall	Lightweight, fast	Small (few MB)	Device/Edge
[20]	2022	Evading malware detection	89.6% defense accuracy	Heavy (robustness overhead)	Large	Cloud
[22]	2023	Random Forest	High accuracy	Moderate	Medium	Cloud
[23]	2024	Hybrid ML + Crypto	99.5% anomaly detection	Moderate (mix of tasks)	Medium	Edge + Cloud
[27]	2021	LSTM IDS	99%	High (sequential RAM use)	Large	Edge/Cloud
[29]	2024	Hybrid CNN-LSTM	97–99%	Very High (needs GPU)	Large (100s MB)	Edge/Cloud
[30]	2023	DRL IDS	Strong (varied datasets)	Extremely High	Very Large	Cloud
[31]	2023	Federated + Transfer Learning	99%	High (distributed training)	Large	Edge + Cloud
[33]	2024	LSTM IDS	97.60%	High	Medium -Large	Edge Server
[34]	2024	CNN + LSTM Hybrid	98.40%	Very High	Large	Edge/Cloud

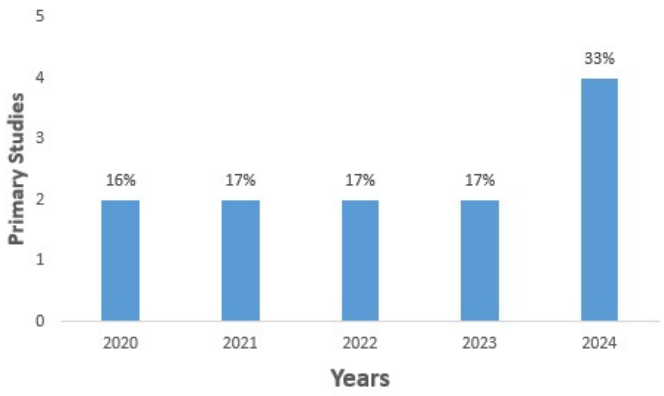


Figure 7. Yearly spread of selected studies.

deployment to cloud or high-capacity edge servers. This creates latency and bandwidth dependencies that may not suit real-time IoT applications. By contrast, lighter ML techniques such as Random Forests or XGBoost achieve slightly lower accuracy but offer faster inference, smaller model sizes, and greater suitability for embedding directly into IoT device firmware or edge gateways. For software engineers, the choice therefore depends on system constraints: if the priority is minimal latency and local autonomy, lightweight ML models are preferable; if the priority is detection depth and adversarial robustness, cloud-integrated DL models are more appropriate [35].

Another synthesis point is lifecycle management. Most reviewed studies optimize for benchmark accuracy but provide little discussion of how models will adapt to real-world change, such as software drift introduced by over-the-air (OTA) firmware updates or heterogeneous IoT environments. Integrating ML/DL into IoT software stacks requires not only strong models but also retraining pipelines, update mechanisms, and compatibility with diverse middleware.

Finally, deployment feasibility is influenced by software dependencies. Studies relying on full DL frameworks (TensorFlow, PyTorch) face challenges in resource-constrained environments, whereas

lightweight runtimes (TensorFlow Lite, ONNX Runtime) remain underutilized in the literature. Bridging this gap between high-performing prototypes and deployable, maintainable software modules is a central challenge for future IoT security research [36].

6 Challenges and Future Directions

Regardless of notable advancements of ML and DL in IoT security, there are certain challenges in applying state-of-the-art techniques. The potential short-term and long-term challenges along with future research directions are given in the following section.

6.1 Short-term challenges

IoT security issues that require immediate attention include weak passwords, unencrypted data transmission, lack of firmware updates, physical tempering, and over-permissive device pairing. Moreover, monitoring gaps, insecure APIs, and vendor backdoors pose serious concerns for IoT security models [1]. These challenges can be addressed by eliminating default credentials, applying security measures on transit data, and ensuring regular firmware updates. Furthermore, hardening insecure APIs, avoiding physical tempering, employing secure device pairing mechanisms, and eliminating vendor backdoors can tackle the most alarming short-term hazards [14].

6.2 Long term challenges

The major hurdle is limited data availability and imbalanced datasets producing overfitted models with poor generalization [37]. Moreover, ML-DL-based models often require high computational power along with significant memory; hence posing difficulties in their deployment on lightweight IoT nodes. Finally, the low generalizability of ML-DL security models leads to inconsistent results restricting the large-scale implementation in real-world applications [38].

Certain challenges are crucial in both short-term and long-term perspectives. For example, attackers can modify incoming data by manipulating the adversarial vulnerability of ML-DL models resulting in misclassified activities [39]. Additionally, IoT networks produce huge amounts of heterogeneous data which demand efficient and low-latency security mechanisms; hence scalability and real-time data processing remain major concerns.

To mitigate all these IoT security challenges, researchers should explore federated learning and edge AI to design lightweight and energy-efficient models while investigating distributed computational workloads and reduced dependency on centralized cloud computing. Moreover, adversarial defense strategies should be designed that integrate robust model training and adversarial detection mechanisms to improve model resistance against evasion attacks [40]. Furthermore, to address data scarcity, researchers need to develop more diverse datasets that represent real-world IoT security datasets and dynamic attack scenarios [41]. Additionally, there is a dire need for cooperation among academia, industry, and regulatory bodies to develop standardized security protocols preserving a streamlined integration of ML/DL-driven approaches. The security challenges faced by IoT devices and integrated environments are presented in Figure 8.

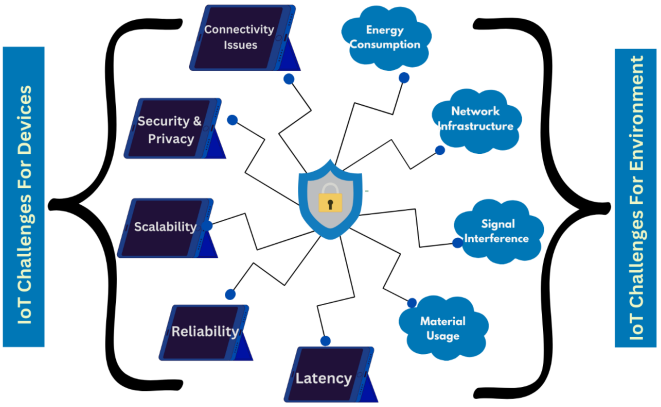


Figure 8. IoT security challenges.

A general comparison of ML and DL models for IoT security is presented in Table 4.

6.3 Adversarial Attacks as a Software Engineering Challenge:

Adversarial examples expose a gap in the software lifecycle of ML-based security systems. From a software engineering perspective, this is a testing and quality assurance challenge: how can we systematically test, validate, and harden security modules against adversarial inputs before deployment? Current research focuses on detection accuracy but rarely considers adversarial robustness within software development pipelines. Future work should integrate adversarial testing into CI/CD workflows, similar to fuzz testing or unit testing in conventional software engineering, ensuring ML-driven IoT security modules remain reliable under evolving attack conditions.

Table 4. General comparison of ML and DL models for IOT security.

Aspect	ML Methods	DL Methods
Accuracy	Moderate to high	Generally high
Latency	Low to moderate	High
Resource Usage	Low (can run on lightweight IoT devices)	High (requires powerful hardware)
Feature Engineering	Manual and domain-specific	Automated hierarchical feature learning
Training Data Requirements	Suitable for smaller datasets	Large-scale datasets required for optimal results
Scalability	Moderate	High
Adaptability to New Threats	Requires retraining	More adaptive
Interpretability	Easier	Difficult
Robustness to Adversarial Attacks	Lower robustness depending upon model	Generally high robustness
Deployment Complexity	Simple deployment	Complex deployment
Energy Consumption	Lower energy footprint	Higher energy consumption
Suitability for Edge Computing	Highly suitable for lightweight deployments	Challenging as needs optimization for edge devices

6.4 Scalability as a Software Architecture Challenge:

Scaling IoT security analytics is not only a data volume problem but also a question of software design. Monolithic IDS architectures often struggle under high throughput, while microservices-based designs allow modular scaling of intrusion detection, anomaly analysis, and logging components. Future research should consider service decomposition, containerization, and orchestration (e.g., Kubernetes) to enable ML/DL security modules to scale flexibly across cloud and edge environments.

6.5 Deployment as a Software Optimization Problem:

Running ML/DL security modules on constrained devices requires more than algorithmic accuracy—it is a software optimization challenge. Techniques such as model quantization, pruning, and compiler-level optimizations can significantly reduce inference latency and memory footprint, enabling lightweight deployment on microcontrollers and embedded firmware. Embedding these optimizations into IoT development pipelines ensures that high-performing models remain deployable in real-world, resource-limited environments.

6.6 Implications for Practice:

For software developers and architects, the review highlights that securing IoT systems with ML/DL requires careful trade-offs between accuracy, resource use, and integration effort. In practice, building an in-house ML-based IDS may only be feasible for organizations with strong data pipelines and ML expertise, while many will find third-party or cloud-based security services more practical. Regardless of source, integration should follow modular principles—treating security analytics as independent services or libraries that can be updated without disrupting the wider IoT stack. Lifecycle considerations are equally critical: ML/DL components must be developed, integrated, monitored, and retrained as IoT devices evolve through OTA updates and changing workloads. Automated pipelines for deployment, adversarial testing, and model updates should therefore be seen as core elements of IoT software engineering, rather than optional add-ons.

7 Conclusion

This review examined recent ML- and DL-based approaches to IoT security (2020–2024) through a software engineering lens. Unlike prior reviews that focus only on detection accuracy, we emphasized deployment models, integration complexity, computational overhead, and software dependencies—factors that directly affect the feasibility of real-world adoption. A key contribution is the synthesis of trade-offs: while deep models such as CNN-LSTM and DRL offer superior accuracy, their

heavy computational demands confine them to cloud or edge platforms; in contrast, lighter methods like Random Forest or XGBoost sacrifice some accuracy but are deployable within device firmware and gateways. We also reframed challenges such as adversarial attacks, scalability, and lifecycle drift as software problems of testing, architecture, and maintainability, offering directions for integrating ML/DL modules into IoT software stacks. For practitioners, the main implication is that secure IoT software must be designed with lifecycle-aware ML components that can be updated, monitored, and optimized over time, rather than treated as static add-ons.

Future work should benchmark ML/DL models on standardized IoT security datasets and evaluate them in deployment-aware terms such as inference time, energy use, and update overhead. Key priorities include building adversarial testing into QA pipelines, adopting scalable software architectures like microservices, and developing lifecycle-aware update pipelines to handle retraining and OTA updates. These steps are essential to move from prototypes to deployable and resilient IoT security software systems.

Data Availability Statement

Not applicable.

Funding

This work was supported without any funding.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Rachakonda, L. P., Siddula, M., & Sathya, V. (2024). A comprehensive study on IoT privacy and security challenges with focus on spectrum sharing in Next-Generation networks (5G/6G/beyond). *High-Confidence Computing, 4*(2), 100220. [Crossref]

[2] Sahu, S. K., & Mazumdar, K. (2024). Exploring security threats and solutions Techniques for Internet of Things (IoT): from vulnerabilities to vigilance. *Frontiers in Artificial Intelligence, 7*, 1397480. [Crossref]

[3] Awad, A. I., Babu, A., Barka, E., & Shuaib, K. (2024). AI-powered biometrics for Internet of Things security: A review and future vision. *Journal of Information Security and Applications, 82*, 103748. [Crossref]

[4] Baral, S., Saha, S., & Haque, A. (2024, November). An Adaptive End-to-End IoT Security Framework Using Explainable AI and LLMs. In *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)* (pp. 469-474). IEEE. [Crossref]

[5] Humayun, M., Tariq, N., Alfayad, M., Zakwan, M., Alwakid, G., & Assiri, M. (2024). Securing the Internet of Things in artificial intelligence era: A comprehensive survey. *IEEE access, 12*, 25469-25490. [Crossref]

[6] Al-Shurbaji, T., Anbar, M., Manickam, S., Hasbullah, I. H., ALfriehate, N., Alabsi, B. A., ... & Hashim, H. (2025). Deep learning-based intrusion detection system for detecting IoT botnet attacks: a review. *IEEE Access*. [Crossref]

[7] Attkan, A., & Ranga, V. (2022). Cyber-physical security for IoT networks: a comprehensive review on traditional, blockchain and artificial intelligence based key-security. *Complex & Intelligent Systems, 8*(4), 3559-3591. [Crossref]

[8] Villegas-Ch, W., Govea, J., Gurierrez, R., & Mera-Navarrete, A. (2025). Optimizing security in IoT ecosystems using hybrid artificial intelligence and blockchain models: a scalable and efficient approach for threat detection. *IEEE Access*. [Crossref]

[9] Al-Garadi, M. A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I., & Guizani, M. (2020). A survey of machine and deep learning methods for internet of things (IoT) security. *IEEE communications surveys & tutorials, 22*(3), 1646-1685. [Crossref]

[10] Desanamukula, V. S., Priyadarshini, M. A., Srilatha, D., Rao, K. V., Kumari, R. L., & Vivek, K. (2023, July). A Comprehensive Analysis of Machine Learning and Deep Learning Approaches towards IoT Security. In *2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC)* (pp. 1165-1168). IEEE. [Crossref]

[11] Chen, F., Luo, D., Li, J., Leung, V. C., Li, S., & Fan, J. (2022). Arm PSA-certified IoT chip security: a case study. *Tsinghua Science and Technology, 28*(2), 244-257. [Crossref]

[12] Xenofontos, C., Zografopoulos, I., Konstantinou, C., Jolfaei, A., Khan, M. K., & Choo, K. K. R. (2021). Consumer, commercial, and industrial iot (in) security: Attack taxonomy and case studies. *IEEE Internet of Things Journal, 9*(1), 199-221. [Crossref]

[13] Rajendran, R. K. (2025). Data Privacy and Security Risks in Third-Party App Integrations. In *Analyzing Privacy and Security Difficulties in Social Media: New Challenges and Solutions* (pp. 311-334). IGI Global Scientific Publishing. [Crossref]

[14] Fazeldehkordi, E., Owe, O., & Noll, J. (2019, May). Security and privacy in IoT systems: a case study of healthcare products. In *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)* (pp. 1-8). IEEE. [Crossref]

[15] Sharma, P., Jain, S., Gupta, S., & Chamola, V. (2021). Role of machine learning and deep learning in securing 5G-driven industrial IoT applications. *Ad Hoc Networks, 123*, 102685. [Crossref]

[16] Sánchez, P. M. S., Celdrán, A. H., Bovet, G., & Pérez, G. M. (2024). Adversarial attacks and defenses on ML-and hardware-based IoT device fingerprinting and identification. *Future Generation Computer Systems, 152*, 30-42. [Crossref]

[17] Haghighi, M. S., Farivar, F., & Jolfaei, A. (2020). A machine-learning-based approach to build zero-false-positive IPSs for industrial IoT and CPS with a case study on power grids security. *IEEE Transactions on Industry Applications, 60*(1), 920-928. [Crossref]

[18] Makkar, A., Garg, S., Kumar, N., Hossain, M. S., Ghoneim, A., & Alrashoud, M. (2020). An efficient spam detection technique for IoT devices using machine learning. *IEEE Transactions on Industrial Informatics, 17*(2), 903-912. [Crossref]

[19] Da Cruz, M. A., Abbade, L. R., Lorenz, P., Mafra, S. B., & Rodrigues, J. J. (2022). Detecting compromised IoT devices through XGBoost. *IEEE transactions on intelligent transportation systems, 24*(12), 15392-15399. [Crossref]

[20] Renjith, G., Vinod, P., & Aji, S. (2022). Evading machine-learning-based Android malware detector for IoT devices. *IEEE Systems Journal, 17*(2), 2745-2755. [Crossref]

[21] Ali, A., Akram, M. A., Farooq, W., Ali, M., Nazir, M., Muhammad, A., & Mazhar, T. (2025). MalwareVison: A Deep Learning-Driven Approach For Malware Classification. *Journal of Computing & Biomedical Informatics, 8*(02).

[22] Chavhan, G. S., Rautkar, A., Prithviraj, J., Agrawal, R., Chavhan, N., & Dhule, C. (2023, November). Machine learning for 5G security using random forest. In *2023 International Conference on Advances in Computation, Communication and Information Technology* (*ICAICCIT*) (pp. 544-549). IEEE. [Crossref]

[23] Subathra, K., Vignesh, G. R., Babu, S. T., Mendhe, D., kumar Yada, R., & Maranan, R. (2024, April). Secure Data Transmission in IoT Networks: A Machine Learning-Based Approach. In *2024 Ninth International Conference on Science Technology Engineering and Mathematics* (*ICONSTEM*) (pp. 1-5). IEEE. [Crossref]

[24] Mahmood, M. A., & Zeki, A. M. (2020, September). Securing IOT against DDOS attacks using machine learning. In *IET Conference Proceedings CP777* (Vol. 2020, No. 6, pp. 471-476). Stevenage, UK: The Institution of Engineering and Technology. [Crossref]

[25] Luqman, M., Zeeshan, M., Riaz, Q., Hussain, M., Tahir, H., Mazhar, N., & Khan, M. S. (2025). Intelligent parameter-based in-network IDS for IoT using UNSW-NB15 and BoT-IoT datasets. *Journal of the Franklin Institute, 362*(1), 107440. [Crossref]

[26] Kumar, P. M., Kavin, B. P., Jagathpally, A., & Shahwar, T. (2025, February). Transforming the cybersecurity space of healthcare IoT devices using Deep Learning. In *2025 IEEE 4th International Conference on AI in Cybersecurity* (*ICAIC*) (pp. 1-6). IEEE. [Crossref]

[27] Koroniotis, N., Moustafa, N., Turnbull, B., Schiliro, F., Gauravaram, P., & Janicke, H. (2021, October). A deep learning-based penetration testing framework for vulnerability identification in internet of things environments. In *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications* (*TrustCom*) (pp. 887-894). IEEE. [Crossref]

[28] Kim, T. H., Srinivasulu, A., Chinthaginjala, R., Dhakshayani, J., Zhao, X., & Obaidur Rab, S. (2025). Enhancing cybersecurity through script development using machine and deep learning for advanced threat mitigation. *Scientific Reports, 15*(1), 8297. [Crossref]

[29] Jablaoui, R., & Liouane, N. (2024, May). An effective deep CNN-LSTM based intrusion detection system for network security. In *2024 International Conference on Control, Automation and Diagnosis* (*ICCAD*) (pp. 1-6). IEEE. [Crossref]

[30] Nguyen, T. T., & Reddi, V. J. (2021). Deep reinforcement learning for cyber security. *IEEE Transactions on Neural Networks and Learning Systems, 34*(8), 3779-3795. [Crossref]

[31] Khoa, T. V., Hoang, D. T., Trung, N. L., Nguyen, C. T., Quynh, T. T. T., Nguyen, D. N., ... & Dutkiewicz, E. (2022). Deep transfer learning: A novel collaborative learning model for cyberattack detection systems in IoT networks. *IEEE Internet of Things Journal, 10*(10), 8578-8589. [Crossref]

[32] Javed, A., Malhi, A., Kinnunen, T., & Främling, K. (2020). Scalable IoT platform for heterogeneous devices in smart environments. *IEEE access, 8*, 211973-211985. [Crossref]

[33] Racherla, S., Sripathi, P., Faruqui, N., Kabir, M. A., Whaiduzzaman, M., & Shah, S. A. (2024). Deep-IDS: A real-time intrusion detector for IoT nodes using deep learning. *IEEE Access, 12*, 63584-63597. [Crossref]

[34] Gueriani, A., Kheddar, H., & Mazari, A. C. (2024, April). Enhancing iot security with cnn and lstm-based intrusion detection systems. In *2024 6th International Conference on Pattern Analysis and Intelligent Systems* (*PAIS*) (pp. 1-7). IEEE. [Crossref]

[35] Dritsas, E., & Trigka, M. (2025). A survey on the applications of cloud computing in the industrial internet of things. *Big data and cognitive computing, 9*(2), 44. [Crossref]

[36] Ray, P. P. (2023). An overview of WebAssembly for IoT: Background, tools, state-of-the-art, challenges, and future directions. *Future Internet, 15*(8), 275. [Crossref]

[37] Mishra, R., & Mishra, A. (2025). Current research on Internet of Things (IoT) security protocols: A survey. *Computers & Security*, 104310. [Crossref]

[38] Ni, C., & Li, S. C. (2024). Machine learning enabled industrial iot security: Challenges, trends and solutions. *Journal of Industrial Information Integration, 38*, 100549. [Crossref]

[39] Halgamuge, M. N., & Niyato, D. (2025). Adaptive edge security framework for dynamic IoT security policies in diverse environments. *Computers & Security, 148*, 104128. [Crossref]

[40] Rehman, Z., Gondal, I., Ge, M., Dong, H., Gregory, M., & Tari, Z. (2024). Proactive defense mechanism: Enhancing IoT security through diversity-based moving target defense and cyber deception. *Computers & Security, 139*, 103685. [Crossref]

[41] Alwahedi, F., Aldhaheri, A., Ferrag, M. A., Battah, A., & Tihanyi, N. (2024). Machine learning techniques for IoT security: Current research and future vision with generative AI and large language models. *Internet of Things and Cyber-Physical Systems, 4*, 167-185. [Crossref]



HAROON ARIF received his Bachelor's degree in Computer Science from Preston University, Islamabad, and his Master's degree in Cybersecurity from the Illinois Institute of Technology, Chicago. He is a cybersecurity professional and researcher with expertise in cloud security, threat intelligence, post-quantum cryptography, and AI-driven threat detection. His work explores the integration of artificial intelligence with cybersecurity to enhance protection against evolving digital threats, particularly in cloud and mobile environments. Haroon is a member of IEEE and actively contributes to the academic community as a peer reviewer for IEEE Access and other scientific journals. He has published multiple papers on topics such as AI-enhanced cloud security, dynamic cryptographic algorithm selection, and resilient enterprise architectures. His research aims to develop innovative, data-driven solutions for securing modern digital infrastructures. (Email: harif@hawk.iit.edu)



Abdul Karim Sajid Ali received his Bachelor's degree in Electronics and Communication Engineering from Osmania University, Hyderabad, and his Master's degree in Information Technology and Management from the Illinois Institute of Technology, Chicago. He is an experienced data engineer and researcher with expertise in cloud data integration, cybersecurity, and AI-driven analytics. His work spans the development of scalable ETL pipelines, secure digital identity systems, and machine learning-based threat detection. Abdul has co-authored several peer-reviewed publications in the areas of post-quantum cryptography, secure mobile communications, and AI-enhanced cybersecurity. He is particularly interested in building resilient data infrastructures and applying intelligent systems to protect sensitive digital environments. His ongoing work focuses on bridging theoretical models with real-world implementation for resilient and scalable digital infrastructures. (Email: aali62@hawk.iit.edu)



Hussain Abdul Nabi is a cybersecurity and artificial intelligence (AI) researcher with a focus on the secure integration of AI and machine learning (ML) into enterprise resource planning (ERP) systems and supply chain operations. He holds a Bachelor of Business Administration from Punjab University and a Master of Business Administration from Superior University, Lahore. With seven years of industry experience in sales, merchandising, and business development, his research is rooted in practical business processes and enterprise risk landscapes. His research interests include cybersecurity in enterprise systems, AI-enabled risk prediction, supply chain resilience, intelligent ERP architectures, and the secure application of machine learning in operational decision-making. His work aligns with global efforts to strengthen digital infrastructures through secure, adaptive, and data-driven technologies. (Email: shhussain024@gmail.com)