iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.81472

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

SafeMesh- Securing The Future of IoT

Hafiz Shamnad, Fathima Hanan, Athul G, Sayanth Sajeev, Amrutha S Aravind, Aswathy L C
Department of Computer Science and Engineering (Cybersecurity) Rajadhani Institute of Engineering and Technology Trivandrum, Kerala, India

Abstract: ***The rapid expansion of the Internet of Things (IoT) ecosystem has fundamentally transformed modern computing environments, enabling seamless connectivity across a wide range of devices including sensors, actuators, mobile systems, and embedded platforms. However, this growth has also introduced significant security challenges arising from device heterogeneity, resource constraints, lack of standardized security mechanisms, and the use of diverse communication protocols such as Wi-Fi, Bluetooth, Zigbee, and USB. Traditional vulnerability assessment techniques predominantly rely on active scanning approaches, which involve direct interaction with target devices through probing, port scanning, or packet injection. While effective in conventional IT infrastructures, these methods are often unsuitable for IoT environments, as they can disrupt device functionality, introduce latency, consume limited device resources, and in some cases trigger unintended system failures. Furthermore, existing solutions frequently lack the capability to provide unified visibility across multiple communication interfaces, resulting in incomplete security assessments.***

To address these limitations, this paper presents SafeMesh, a passive IoT vulnerability scanning and analysis framework designed to provide comprehensive security visibility without interfering with normal device operations. The proposed system leverages passive monitoring techniques, including ARP observation, protocol inspection, and metadata extraction, to identify and profile devices across heterogeneous interfaces such as IP-based networks, Bluetooth, and USB connections. SafeMesh incorporates protocol-aware device fingerprinting mechanisms to accurately classify devices and infer their characteristics, enabling more precise vulnerability mapping. The framework integrates external threat intelligence sources, including standardized databases such as the Common Vulnerabilities and Exposures (CVE) and the National Vulnerability Database (NVD), to correlate detected devices with known security weaknesses.

A key contribution of this work is the introduction of a layered intelligence model that systematically processes collected data through stages of enrichment, vulnerability correlation, and contextual risk assessment. The risk evaluation mechanism employs a weighted scoring model that considers multiple factors, including device exposure, vulnerability severity, configuration weaknesses, and operational criticality, thereby producing normalized risk scores for effective prioritization. In addition, SafeMesh incorporates a digital twin-based simulation environment that creates virtual representations of networked devices, enabling the execution of controlled attack scenarios such as lateral movement, malware propagation, and denial-of-service attacks. This simulation capability allows for predictive analysis of potential threats and evaluation of network resilience without impacting the live environment.

The proposed approach offers several advantages, including reduced network overhead, continuous monitoring capability, and improved scalability for deployment in edge and resource-constrained environments. Experimental evaluation conducted in a controlled IoT testbed demonstrates that SafeMesh can accurately identify devices, detect associated vulnerabilities, and generate actionable security insights with minimal latency and resource consumption. The results further indicate that passive scanning, when combined with contextual intelligence and simulation-driven analysis, provides a viable and efficient alternative to traditional active vulnerability assessment techniques.

Overall, SafeMesh contributes to advancing IoT security by integrating passive discovery, multi-interface visibility, threat intelligence correlation, and predictive simulation into a unified framework. The system enhances situational awareness, supports proactive risk management, and provides a scalable solution for securing complex and dynamic IoT ecosystems.

Keywords: IoT Security, Passive Scanning, Vulnerability Assessment, CVE, Network Monitoring, Digital Twin, Risk Analysis

I. INTRODUCTION

The rapid adoption of IoT technologies has led to the deployment of interconnected devices across critical sectors such as healthcare, smart homes, industrial automation, and education. These environments consist of devices with varying computational capabilities, communication protocols, and security configurations, creating a highly fragmented attack surface. A key challenge in IoT security lies in achieving unified visibility across multiple communication layers.

Existing tools often operate within isolated domains (e.g., network-only scanning), resulting in incomplete asset identification and fragmented security insights. Furthermore, active scanning techniques introduce additional traffic, which may disrupt latency-sensitive or resource-constrained IoT devices.

SafeMesh addresses these limitations by proposing a passive, multi-interface security assessment framework. Instead of injecting traffic, the system observes device behaviour, extracts protocol-level metadata, and correlates findings with known vulnerabilities. This approach ensures minimal operational impact while providing a comprehensive view of the IoT environment.

II. RELATED WORKS

Existing research in IoT security primarily focuses on three areas: authentication mechanisms, vulnerability detection, and attack modeling. Studies on authentication protocols emphasize lightweight cryptographic solutions for resource-constrained devices but often lack integration with real-time monitoring systems. While these approaches strengthen access control, they do not address post-deployment vulnerability assessment.

Attack graph-based models provide a structured method for analyzing potential attack paths by mapping interdependencies between devices. However, these models become computationally expensive as network size increases and often require complete knowledge of system topology, which is not always feasible in dynamic IoT environments. Machine learning-based detection systems have been proposed for anomaly detection and threat prediction. Although effective in identifying unknown threats, they require large datasets and training overhead, limiting their applicability in real-time edge deployments.

In contrast, SafeMesh focuses on practical deployment feasibility, combining passive monitoring with lightweight analysis techniques. Instead of relying solely on predictive models, it leverages established vulnerability databases and contextual intelligence to provide actionable insights.

III. EXISTING SYSTEM ANALYSIS

Current IoT security solutions exhibit several strengths but also significant limitations when applied to heterogeneous environments. Traditional vulnerability scanners rely on signature-based detection, enabling accurate identification of known threats. However, these systems are typically constrained to specific protocols or network layers, resulting in partial visibility. Another limitation is the dependence on active probing techniques, which can introduce performance degradation and unintended disruptions. This is particularly problematic in environments such as healthcare or industrial control systems where stability is critical.

Additionally, existing tools often lack integrated risk prioritization mechanisms. Vulnerabilities are presented as isolated findings without contextual evaluation, making it difficult for administrators to identify high-impact threats. SafeMesh differentiates itself by addressing these gaps through:

- Passive observation instead of active probing
- Cross-interface device discovery
- Context-aware risk scoring
- Centralized and structured reporting

IV. PROPOSED SYSTEM

SafeMesh is designed as a centralized IoT security node that continuously monitors device activity and performs vulnerability analysis without disrupting network operations. The system aggregates data from multiple interfaces and processes it through a structured pipeline consisting of discovery, enrichment, analysis, and reporting stages. The system introduces several capabilities that extend beyond traditional scanning tools:

- Multi-Interface Discovery: Detects devices across heterogeneous communication channels, ensuring complete asset visibility.
- Passive Monitoring: Observes network traffic and device metadata without injecting packets, preserving system stability.
- Contextual Intelligence: Enhances raw scan data with vendor information, service identification, and vulnerability mapping.
- Risk Prioritization: Assigns severity levels based on multiple factors, enabling efficient threat mitigation.
- Simulation Support: Models potential attack scenarios using digital twins to evaluate system resilience.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

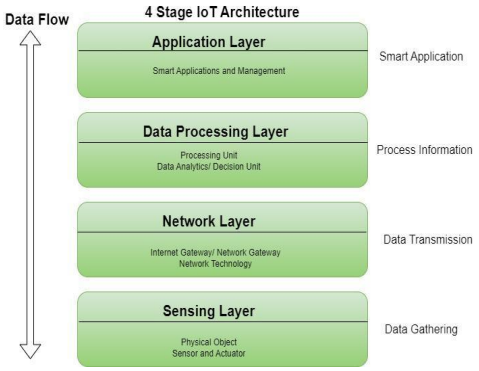


Figure 1: 4 stage of IoT Architecture

Figure 1 presents the layered architecture of the SafeMesh framework, adapted from the conventional IoT model and enhanced with security-centric components. At the lowest level, the Scanner Layer replaces the traditional sensing layer by performing passive device discovery using techniques such as ARP observation and protocol inspection. The collected data is transmitted through the communication layer and processed within the Intelligence Layer, where enrichment, vulnerability correlation, and risk scoring are performed. Above this, the Application Layer manages backend services and API interactions, enabling seamless integration between modules. Finally, the User Layer provides an interactive dashboard for monitoring, visualization, and control. This layered abstraction ensures a structured data flow from raw device acquisition to actionable security insights, improving scalability, modularity, and analytical efficiency.

V. SYSTEM ARCHITECTURE

The proposed architecture adopts a modular, layered design paradigm to ensure high cohesion within components and low coupling between them, enabling independent development, scalability, and easier maintenance. At the topmost level, the User Layer serves as the primary interface, offering an interactive dashboard through which users can visualize system states, monitor security events, and initiate control actions. This layer is designed with usability and responsiveness in mind, ensuring real-time feedback and intuitive navigation. Supporting this is the Application Layer, which acts as the orchestration core of the system. It manages business logic, coordinates inter-layer communication, and exposes APIs that facilitate seamless integration between frontend interfaces and backend services.

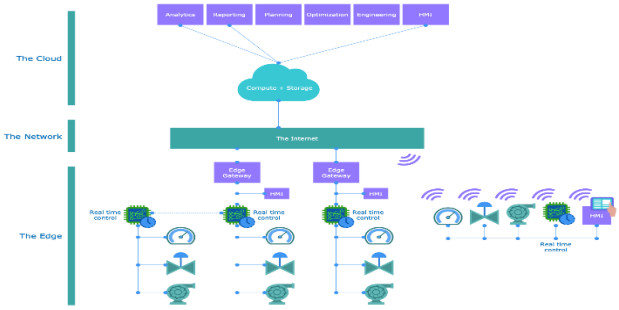


Figure 2: Typical Industrial Internet of Things (IIoT) architecture

Figure 2 illustrates a typical Industrial Internet of Things (IIoT) architecture, structured into three primary layers: Edge, Network, and Cloud. The Edge layer consists of physical devices such as sensors, actuators, and embedded controllers that generate real-time data and perform localized control operations. These devices communicate through edge gateways, which act as intermediaries for data aggregation and protocol translation. The Network layer facilitates data transmission across the Internet, enabling connectivity between edge devices and centralized systems. At the top, the Cloud layer provides computational resources, storage, and advanced services such as analytics, reporting, optimization, and human-machine interfaces (HMI). This layered model highlights the distributed and heterogeneous nature of IoT environments, where devices operate across multiple interfaces and communication protocols. Such complexity introduces significant security challenges, particularly in achieving unified visibility and vulnerability assessment across all layers, thereby motivating the need for frameworks such as SafeMesh.

Beneath the application logic, the Scanner Layer is responsible for continuous data acquisition from network nodes, IoT devices, and system endpoints. It performs active and passive scanning to gather raw telemetry, configuration details, and potential vulnerability indicators. This unprocessed data is forwarded to the Intelligence Layer, where advanced processing techniques—such as rule-based analysis, anomaly detection, and threat intelligence correlation—are applied to transform raw inputs into meaningful, actionable insights. This layer is critical for identifying patterns, detecting potential threats, and enriching data with contextual security information.

The Data Layer functions as the persistence backbone of the architecture, storing structured and semi-structured data in optimized databases. It supports efficient querying, historical analysis, and reporting, ensuring data integrity and availability. Parallel to this, the Authentication Layer enforces strict access control policies using token-based authentication mechanisms such as JWT, ensuring that only authorized users and services can interact with the system. It also supports role-based access control (RBAC), session management, and secure credential handling.

A distinctive feature of this architecture is the inclusion of the Digital Twin Layer, which creates dynamic virtual replicas of physical devices and systems. These digital twins mirror real-time states and behaviors, enabling simulation, monitoring, and predictive diagnostics without impacting the actual environment. Building upon this, the Simulation Layer facilitates the execution of controlled attack scenarios, penetration testing models, and "what-if" analyses. This allows the system to evaluate potential vulnerabilities, assess defense mechanisms, and proactively prepare for emerging threats.

Finally, the Reporting Layer consolidates processed data and analytical outputs into structured reports, dashboards, and visualizations. It supports automated report generation, customizable metrics, and exportable formats for stakeholders, aiding in strategic decision-making and compliance requirements.

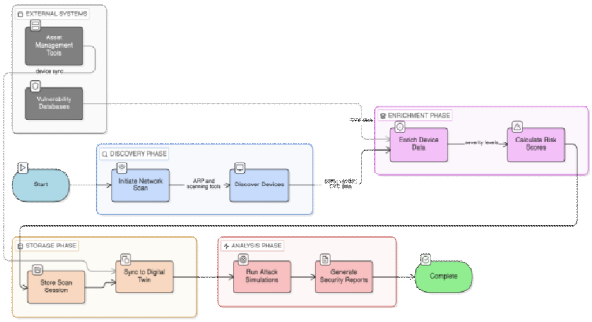


Figure 3: System Architecture of the SafeMesh

Figure3 illustrates the overall system architecture of the SafeMesh framework, structured as a sequential pipeline of interconnected phases that enable efficient IoT device discovery, analysis, and risk evaluation. The process begins with the Discovery Phase, where a network scan is initiated using ARP and related scanning techniques to identify connected devices. This raw device information is then forwarded to the Enrichment Phase, where external systems such as asset management tools and vulnerability databases contribute additional context, enabling device data enrichment and calculation of risk scores based on severity levels. The enriched data flows into the Storage Phase, where scan sessions are securely stored and synchronized with a digital twin representation, ensuring persistent and structured data management. Following this, the Analysis Phase leverages the stored and enriched data to run attack simulations and generate detailed security reports, providing insights into vulnerabilities and potential threats. The architecture concludes with a completion stage, indicating the end of the processing pipeline. Overall, the figure highlights a modular and data-driven workflow, integrating external intelligence sources with internal processing layers to deliver continuous monitoring, predictive analysis, and actionable security outcomes.

Overall, this layered architecture ensures a streamlined data flow pipeline—from data acquisition and processing to simulation and reporting—while maintaining flexibility, extensibility, and robustness. Each layer operates as an independent yet interconnected module, allowing the system to evolve efficiently in response to new requirements, technologies, and threat landscapes.

VI. METHODOLOGY

The methodology adopted in this work follows a structured, multi-stage pipeline designed to ensure efficient data collection, intelligent processing, and accurate risk evaluation within IoT environments.

The process begins with passive data acquisition, where device discovery is carried out using techniques such as ARP observation, protocol inspection, and metadata extraction, minimizing network disruption while maintaining continuous visibility. To extend coverage beyond traditional network interfaces, additional tools are integrated to monitor Bluetooth and USB communications, enabling comprehensive device identification across heterogeneous environments.

Once the raw data is collected, it undergoes enrichment to enhance its analytical value. This includes identifying device vendors through MAC address mapping, detecting active services using port analysis, and classifying devices via fingerprinting techniques. These enrichment steps provide contextual understanding, transforming low-level data into structured, meaningful information. Following this, vulnerability correlation is performed by querying established vulnerability databases to identify known security issues associated with detected devices. Each identified vulnerability is mapped to standardized severity metrics and contextualized based on the specific characteristics and role of the device within the network.

Subsequently, a risk assessment model is applied to evaluate the overall security posture of each device. This model incorporates multiple weighted factors, including exposure level (such as open ports and running services), severity of known vulnerabilities (e.g., CVE scores), device criticality within the network, and potential configuration weaknesses. The computed risk values are then normalized and categorized into defined severity levels, enabling clear prioritization of threats. Overall, this methodology ensures a balanced approach that combines passive monitoring, contextual intelligence, and systematic risk evaluation to deliver accurate and actionable security insights.

VII. IMPLEMENTATION

The implementation of the SafeMesh framework is built upon a lightweight yet powerful technology stack that supports scalability, performance, and ease of integration. The backend services are developed using FastAPI, which provides high-performance API handling and asynchronous capabilities, making it suitable for real-time data processing. For data persistence, SQLite is utilized as a compact and efficient database solution, ensuring structured storage and quick retrieval of device and vulnerability information. Network analysis is conducted using tools such as Nmap and Scapy, which enable detailed inspection of network traffic and device characteristics, while BlueZ is integrated to facilitate Bluetooth scanning and analysis.

To ensure robust security, the system incorporates a multi-layered authentication mechanism. User credentials are protected through secure password hashing using bcrypt, while an additional layer of security is provided via OTP-based two-factor authentication. Session management is handled using JWT tokens, which enable secure and stateless communication between clients and the backend. This layered security approach ensures that only authorized users can access sensitive system functionalities and data.

The overall system integration is achieved through RESTful APIs and WebSocket communication. REST APIs facilitate modular interaction between different system components, allowing independent development and scalability, while WebSockets enable real-time updates for monitoring dashboards and alerts. This hybrid communication model ensures both reliability and responsiveness, making the system adaptable to dynamic IoT environments.

Algorithm 1: Passive IoT Device Discovery and Risk Evaluation

Input: Network traffic stream T, vulnerability database V

Output: Device list D with associated risk scores

1) Initialize empty device set $D = \{\}$

2) Capture network traffic passively from interfaces (WiFi, Bluetooth, USB)

3) For each packet $p \in T$:

- Extract metadata (MAC, IP, protocol, ports)
- Identify unique devices and add to D

4) For each device $d \in D$:

- Perform device fingerprinting (vendor, OS, type)
- Identify active services and open ports

5) Enrich device data using MAC vendor mapping and protocol analysis

6) Query vulnerability database V to retrieve known CVEs for device d

7) For each vulnerability v:

- Extract severity score (CVSS)

8) Compute risk score R_d using weighted model:

$$\boldsymbol{R_d = w_1 E + w_2 V + w_3 C + w_4 W}$$

Where:

E = Exposure level (ports/services)

V = Vulnerability severity (CVSS)

C = Device criticality

W = Configuration weakness

w_i = weights

9) Normalize R_d to range $[0, 10]$

10) Classify risk level: Low, Medium, High, Critical

11) Store results and generate report

The risk assessment process in SafeMesh is formalized through a passive IoT device discovery and evaluation algorithm. The algorithm operates by capturing network traffic across multiple interfaces and extracting device-specific metadata without active probing. Each identified device undergoes fingerprinting and service analysis, followed by enrichment using vendor mapping and protocol characteristics. The system then correlates detected devices with known vulnerabilities from standardized databases such as CVE and NVD. A weighted risk scoring model is applied, incorporating factors such as exposure level, vulnerability severity, device criticality, and configuration weaknesses. The computed risk scores are normalized and categorized into severity levels, enabling effective prioritization of security threats. This algorithm ensures a systematic and scalable approach to IoT security assessment while maintaining minimal operational overhead.

VIII. RESULTS AND EVALUATION

The SafeMesh system was evaluated within a controlled IoT test environment comprising multiple device types, including network-connected and short-range communication devices. The evaluation focused on assessing device discovery accuracy, vulnerability detection capability, system latency, and overall operational impact. The results demonstrated that SafeMesh successfully identified a wide range of devices across different interfaces, including network, Bluetooth, and USB, while accurately detecting associated vulnerabilities with minimal delay.

A key advantage observed during evaluation was the effectiveness of the passive scanning approach. By avoiding intrusive probing techniques, the system significantly reduced network overhead and prevented potential disruptions to device operations. This ensured stable performance even in environments with resource-constrained IoT devices. Additionally, the passive methodology enabled continuous monitoring, allowing the system to detect changes in the network in real time without affecting normal functionality.

The reporting module further enhanced the system's usability by providing clear and structured insights into device inventories, vulnerability distributions, and risk levels. These insights allowed for efficient prioritization of security measures, enabling administrators to focus on high-risk devices and critical vulnerabilities. Overall, the evaluation confirms that SafeMesh delivers reliable performance, accurate detection, and operational efficiency in IoT security monitoring.

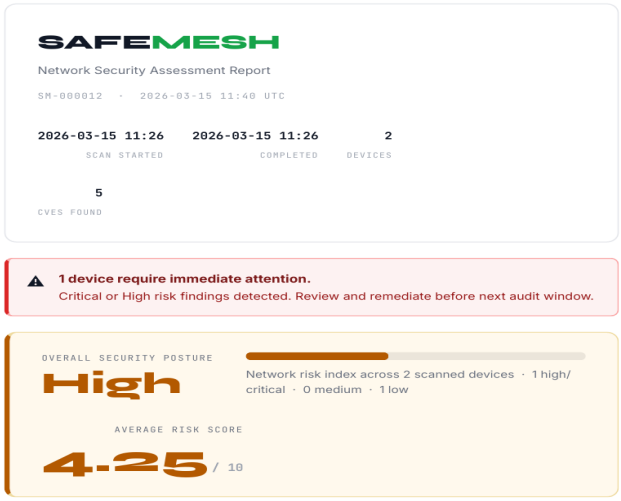


Figure 4 : Generated Report

Figure 4 presents the SafeMesh network security assessment report generated after a passive scan of the target environment. The report indicates that a total of two devices were identified during the scan, with five associated vulnerabilities (CVEs) detected. Among these, one device is classified as high risk, triggering an alert that requires immediate attention. The overall security posture of the network is categorized as High Risk, with an average risk score of 4.25/10, reflecting the presence of significant security concerns. The report also provides a concise risk distribution, showing one high-risk device and one low-risk device, with no medium-risk findings. This figure demonstrates the system's capability to aggregate scan results, evaluate risk levels, and present actionable insights in a clear and structured format for effective decision-making.

IX. FUTURE WORK

Future work will focus on extending the capabilities of the SafeMesh framework to address current limitations and enhance its adaptability to evolving security requirements. One key direction is the integration of real-time anomaly detection mechanisms, which would enable the system to identify unknown threats and abnormal behaviours beyond predefined vulnerability signatures. Additionally, expanding support for a wider range of IoT protocols will improve compatibility with diverse devices and communication standards.

Another important enhancement involves enabling distributed deployment, allowing the system to scale effectively across large and complex network infrastructures. This would facilitate decentralized monitoring and improve performance in enterprise-level environments. Furthermore, incorporating machine learning-based risk prediction models could significantly enhance the accuracy and intelligence of risk assessment, enabling proactive security management. These advancements will further strengthen SafeMesh as a comprehensive and future-ready IoT security solution.

X. CONCLUSION

This paper presented SafeMesh, a passive IoT vulnerability scanning and analysis framework designed to address the growing challenges of securing heterogeneous and resource-constrained IoT environments. Unlike traditional active scanning approaches, which may introduce network overhead, disrupt device functionality, or fail to provide comprehensive multi-interface visibility, SafeMesh adopts a passive monitoring strategy that enables continuous and non-intrusive security assessment. By leveraging techniques such as protocol-aware device fingerprinting, metadata extraction, and multi-interface data acquisition across IP-based networks, Bluetooth, and USB, the framework ensures broad and unified visibility of connected devices.

A key strength of the proposed system lies in its modular and layered architecture, which integrates device discovery, data enrichment, vulnerability intelligence, and risk assessment into a cohesive pipeline. The incorporation of external threat intelligence sources, including standardized databases such as CVE and NVD, enables accurate correlation of detected devices with known vulnerabilities. Furthermore, the weighted risk assessment model provides a structured mechanism for evaluating device security by considering multiple contextual factors such as exposure level, vulnerability severity, configuration weaknesses, and operational criticality. This results in normalized risk scores that facilitate effective prioritization and decision-making.

Another notable contribution of SafeMesh is the integration of a digital twin-based simulation environment, which extends the system's capabilities beyond detection to predictive security analysis. By creating virtual representations of devices and network topologies.

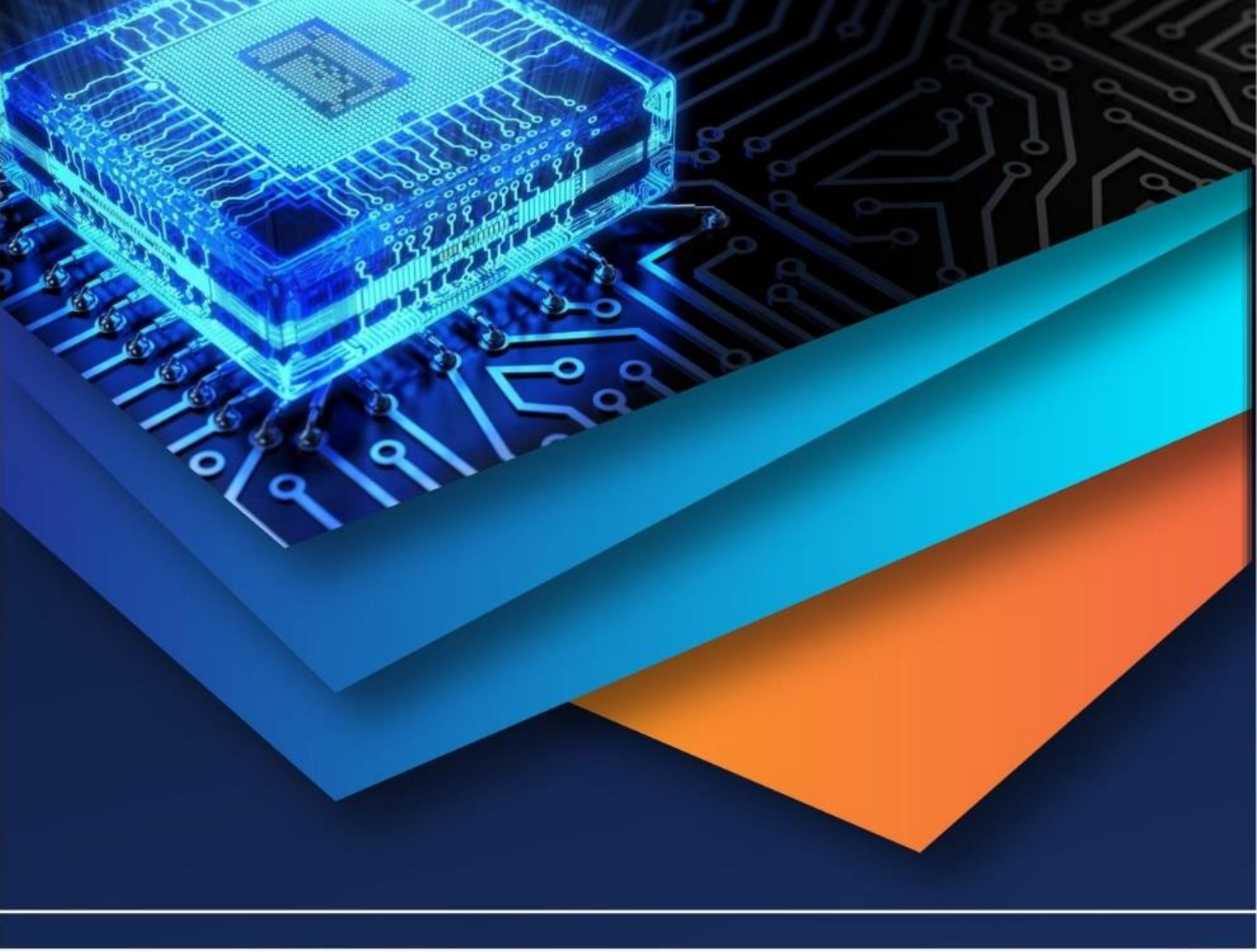
In summary, SafeMesh offers a practical, scalable, and efficient solution for IoT security management by combining passive monitoring, contextual intelligence, and predictive simulation within a unified framework. The system not only improves visibility and risk assessment but also supports proactive defense strategies, thereby contributing to the advancement of secure and resilient IoT ecosystems., correlates vulnerabilities, and provides actionable insights for improving IoT security posture. Its modular architecture and lightweight implementation make it suitable for real-world deployment, particularly in environments where performance and stability are critical. Overall, SafeMesh represents a practical and efficient approach to addressing the growing challenges of IoT security management

REFERENCES

[1] A. Makhshari and A. Mesbah, "IoT bugs and development challenges," in Proc. IEEE/ACM 43rd Int. Conf. Softw. Eng. (ICSE), pp. 460–472, 2021.
[2] B. Zhao et al., "A large-scale empirical analysis of the vulnerabilities introduced by third-party components in IoT firmware," in Proc. 31st ACM SIGSOFT Int. Symp. Softw. Testing Anal., pp. 442–454, 2022.
[3] A. Al-Boghdady, K. Wassif, and M. El-Ramly, "The presence, trends, and causes of security vulnerabilities in operating systems of IoT's low-end devices," Sensors, vol. 21, no. 7, p. 2329, 2021.

[4] X. Jiang, M. Lora, and S. Chattopadhyay, "An experimental analysis of security vulnerabilities in industrial IoT devices," ACM Trans. Internet Technol., vol. 20, no. 2, pp. 1–24, 2020.
[5] Z. B. Celik, E. Fernandes, E. Pauley, G. Tan, and P. McDaniel, "Program analysis of commodity IoT applications for security and privacy: Challenges and opportunities," ACM Comput. Surv., vol. 52, no. 4, pp. 1–30, 2019.
[6] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, "Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations," IEEE Commun. Surveys Tuts., vol. 21, no. 3, pp. 2702–2733, 3rd Quart., 2019.
[7] K. Chen et al., "Internet of Things security and vulnerabilities: Taxonomy, challenges, and practice," J. Hardw. Syst. Secur., vol. 2, pp. 97–110, Jun. 2018.
[8] B. Ali and A. I. Awad, "Cyber and physical security vulnerability assessment for IoT-based smart homes," Sensors, vol. 18, no. 3, p. 817, 2018.
[9] Z. Ling, J. Luo, Y. Xu, C. Gao, K. Wu, and X. Fu, "Security vulnerabilities of Internet of Things: A case study of the smart plug system," IEEE Internet Things J., vol. 4, no. 6, pp. 1899–1909, Dec. 2017.
[10] M. Antonakakis et al., "Understanding the Mirai botnet," in Proc. 26th USENIX Secur. Symp,pp. 1093–1110, 2017.
[11] F. Samie, L. Bauer, and J. Henkel, "IoT technologies for embedded computing: A survey," in Proc. 11th IEEE/ACM/IFIP Int. Conf. Hardw./Softw. Codesign Syst. Synth., 2016.
[12] M. S. Mahmoud and A. A. Mohamad, "A study of efficient power consumption wireless communication techniques/modules for Internet of Things (IoT) applications," Adv. Internet Things, vol. 6, no. 2, pp. 19–29, 2016.
[13] P. P. Ray, "A survey of IoT cloud platforms," Future Comput. Inform. J., vol. 1, nos. 1–2, pp. 35–46, 2016.
[14] J. Ahamed and A. V. Rajan, "Internet of Things (IoT): Application systems and security vulnerabilities," in Proc. 5th Int. Conf. Electron. Devices, Syst. Appl. (ICEDSA), 2016.
[15] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," IEEE Commun. Surveys Tuts., vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)