An Energy-Efficient ESP32 IoT System for Real-Time Detection of WiFi Deauthentication Attacks

Faizal Riza
Department of Informatics Engineering, Institut Teknologi Budi Utomo, Jakarta, Indonesia

Dannie Febrianto Hendrakusuma
Department of Informatics Engineering, Institut Teknologi Budi Utomo, Jakarta, Indonesia

Budi Wibowo
Department of Informatics Engineering, Institut Teknologi Budi Utomo, Jakarta, Indonesia

Abstract: WiFi deauthentication attacks pose a serious threat to users on public WiFi networks by forcibly disconnecting them from access points, often as a prelude to man-in-the-middle exploits. To counter this threat, we developed an energy-efficient ESP32-based IoT system that monitors WiFi traffic in real time to identify deauthentication attack patterns. The device captures deauthentication frames in monitor mode and immediately notifies users through on-device audible/visual alarms (buzzer, LED/OLED) and digital channels (MQTT dashboard and Telegram bot). Experimental evaluation under moderate and high attack scenarios demonstrated robust performance: detection accuracy remained above 97% even under heavy attack traffic (97.8% at peak intensity). Furthermore, the system's duty-cycled design limited average power consumption to ~79 mA (~30% lower than continuous monitoring) and achieved a rapid notification latency of ~270 ms, confirming real-time responsiveness. By combining physical indicators with online alerts, the system effectively warns users and improves public digital security literacy by making cyber threats immediately visible and understandable. Overall, these results establish the proposed system as a low-power, real-time attack detection solution that enhances WiFi network security and user awareness.

Keywords: WiFi Network Security, Deauthentication Attacks, IoT ESP32, Real-Time Detection.

Correspondents Author:
Faizal Riza, Department of Informatics Engineering, Institut Teknologi Budi Utomo, Jakarta, Indonesia
Email: faizalriza@itbu.ac.id

Received October 8 2025; Revised October 31, 2025; Accepted November 1, 2025; Published November 3, 2025.

Introduction

In this ever-evolving digital age, wireless network protection is crucial in maintaining data security and user privacy. One of the most common threats to WiFi networks is the deauthentication attack, which cybercriminals use to sever the connection between a user's device and the access point (Addison et al., 2025). This attack is often the first step in network exploitation, such as Evil Twin or Man-in-the-Middle (MitM) attacks, which can lead to the theft of sensitive information (Schepers et al., 2022).

Currently, many organizations and individuals are still unaware of the threat of deauthentication and how to detect it effectively. Conventional approaches, such as the use of firewalls or network encryption, are often insufficient in dealing with these attacks because deauthentication exploits weaknesses in the IEEE 802.11 protocol, particularly in the authentication and connection management systems. Therefore, an early detection system is needed that can recognize these attacks in real-time and alert users or network administrators. Data from the Indonesian Internet Service Providers Association (APJII) indicates that internet usage in Indonesia has increased from 66.48% of the population in 2022 to 79.5% or 221 million users in 2024, maIn this ever-evolving digital age, wireless network protection is crucial in maintaining data security and user privacy. One of the most common threats to WiFi networks is the deauthentication attack, which cybercriminals use to sever the connection between a user's device and the access point. This attack is often the first step in network exploitation, such as Evil Twin or Man-in-the-Middle (MitM) attacks, which can lead to the theft of sensitive information (Arreaga et al., 2023).

Currently, many organizations and individuals are still unaware of the threat of deauthentication and how to detect it effectively (Zelle et al., 2022). Conventional approaches, such as the use of firewalls or network encryption, are often insufficient in dealing with these attacks because deauthentication exploits weaknesses in the IEEE 802.11 protocol, particularly in the authentication and connection management systems. Therefore, an early detection system is needed that can recognize these attacks in real-time and alert users or network administrators (Litayem & Al-Sa'di, 2023). The world is undergoing a massive digital transformation across sectors (Gebresilassie et al., 2023). This growth drives a sharp rise in WiFi usage for personal, business, and public activities, accompanied by increasing cybersecurity risks (Park et al., 2021). Among these, deauthentication attacks which exploit vulnerabilities in the IEEE 802.11 protocol to forcibly disconnect users are particularly concerning because they can serve as entry points for Evil Twin or Man-in-the-Middle (MitM) attacks that compromise privacy and service continuity (Yazdinejad et al., 2021).

A critical review of existing literature reveals that current deauthentication attack detection methods are ill-suited for low-power, portable, and real-time public deployment. Prior studies have largely depended on high-power computing systems. These approaches typically involve laptop-based packet analyzers using tools like Scapy or sophisticated detectors that analyze multiple parameters like reason codes and RSSI on a full computer (Addison et al., 2025). Other research has focused on resource-intensive, machine-learning-based intrusion detection systems that require complex configurations and significant computing power. While these solutions prove effective in controlled, high-resource environments, their fundamental limitations being resource-intensive, expensive, and complex to configure render them impractical for low-power IoT applications (Harkai, 2024). This analysis identifies a significant research gap: there is no lightweight, energy-efficient, and client-side IoT solution capable of providing accessible, real-time deauthentication detection and alerts directly to non-technical users (Wibowo & Yuswanto, 2023).

To address the identified gaps, this research introduces a new and efficient IoT-based approach. Our system uniquely integrates the ESP32 promiscuous mode with a duty-cycled channel-hopping mechanism, enabling it to efficiently capture and identify malicious deauthentication frames with minimal processing overhead. The main novelty lies in its hybrid alert strategy: the system combines instant physical alerts (LED and buzzer) with powerful digital notifications (MQTT dashboard and Telegram bot). This novel combination serves as an innovative strategy to bridge the gap between real-time threat detection and direct user awareness. The system's effectiveness is validated by evaluating key performance metrics, specifically detection accuracy, notification latency, and overall power consumption. These results confirm the novelty and practicality of the system, demonstrating a significant contribution that goes beyond replication by offering a new, efficient, and accessible IoT-based security solution.

Research Method

The process flow of WiFi deauthentication attack detection using ESP32-based energy-efficient IoT module for real-time detection of deauthentication attacks on Wi-Fi networks in improving digital security literacy is presented in Figure 1.

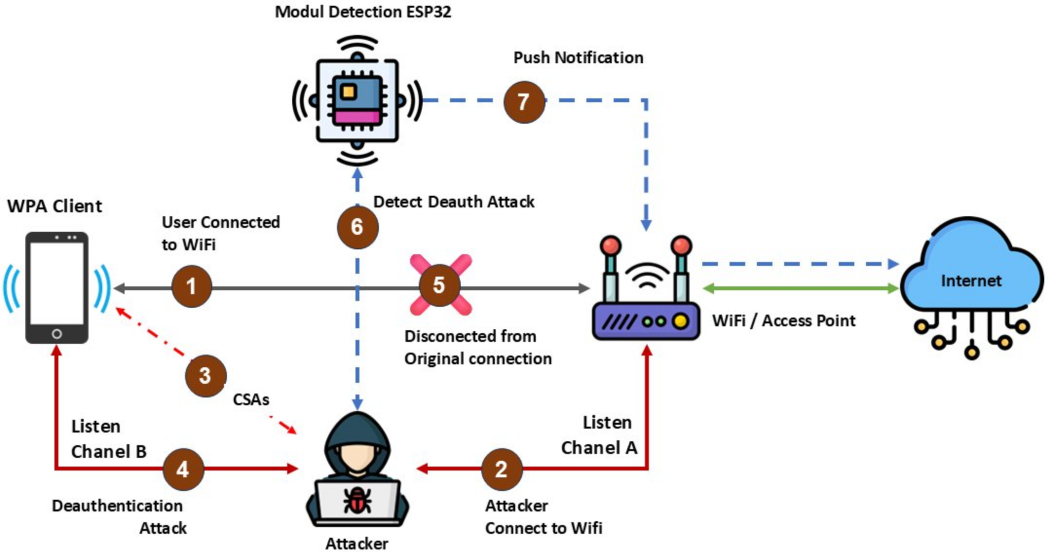


Figure 1 Process flow of WiFi deauthentication attack detection using ESP32-based IoT module

This research method was systematically designed to achieve the predetermined objectives. The research began with the system design stage, which included the selection of hardware and software to be used (Selvarathinam et al., 2019). ESP32 was chosen as the main component in the deauthentication attack detection system due to its capabilities in monitor mode and ease of implementation at low cost. The next stage was the development of software to detect deauthentication packets sent over a WiFi network (Zahid et al., 2023). Programming was carried out using C and MicroPython languages, and initial testing was conducted in a laboratory environment to ensure detection accuracy. After the software was developed, the system was tested with various deauthentication attack scenarios to measure its effectiveness in real conditions (Allafi & Alzahrani, 2024). The results of these tests will be analyzed to assess the reliability of the system in detecting attacks and providing real-time alerts to users (Soner et al., 2024;Padhy et al., 2023). The achievement indicators used include the detection success rate, system response speed, and device power consumption. During the research, flowcharts were used to illustrate the steps that had been and would be taken. These diagrams provided a clear picture of the research process, from the planning stage to the evaluation of results. With this systematic research method, it is hoped that the research will produce an effective, economical ESP32-based deauthentication attack detection system that can be applied by the general public (Ahadi et al., 2020).

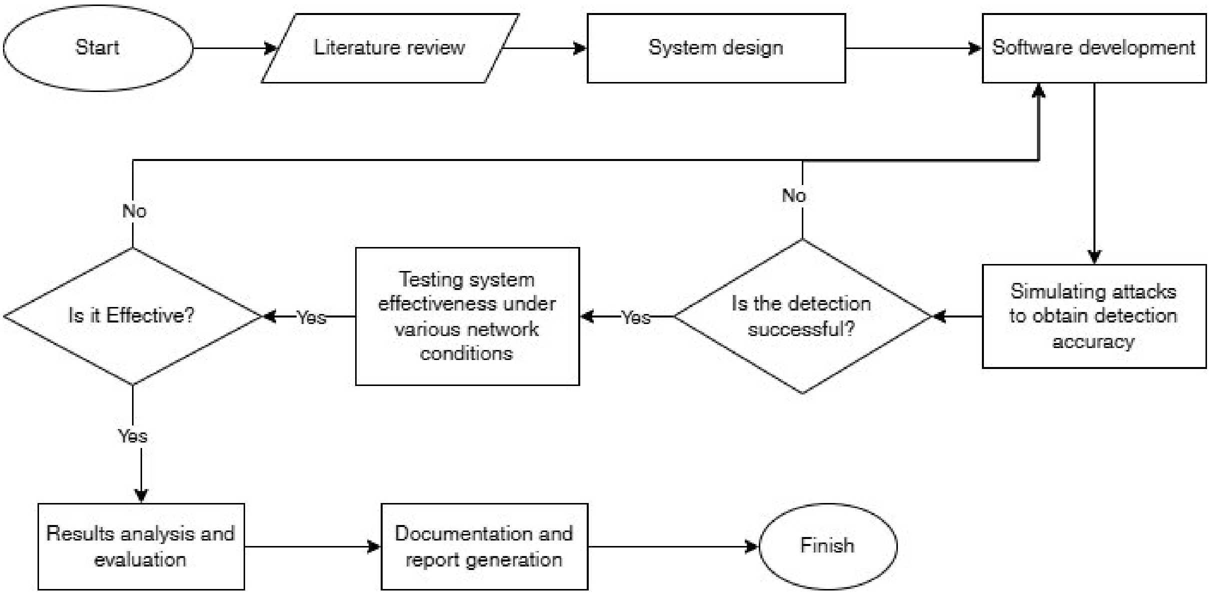

Start
Literature review
System design
Software development
Simulating attacks to obtain detection accuracy
Is the detection successful?
No
Yes
Testing system effectiveness under various network conditions
Yes
Is it Effective?
No
Yes
Results analysis and evaluation
Documentation and report generation
Finish

Figure 2 Stages of deauthentication attack detection research methodology

This study employed an applied experimental research approach to design and evaluate an energy efficient IoT based system for real-time detection of WiFi deauthentication attacks. The methodology was structured to ensure analytical rigor, technical reliability, and reproducibility of results. The overall research framework is illustrated in Fig. 2, which presents the systematic relationship between input variables, processes, and outputs. The research began with the system design phase, which defined the hardware and software architecture. The ESP32 microcontroller was selected as the core component due to its integrated WiFi module, low power consumption, and capability to operate in promiscuous mode, allowing real-time capture of IEEE 802.11 management frames. Supporting components, including an OLED display, a buzzer, and an RGB LED, were integrated to provide immediate visual and auditory alerts. A block diagram was developed to model the data flow and interaction among components (Permana et al., 2023).

During the software development stage, programming was implemented using C and MicroPython within the Arduino IDE environment to enable efficient low-level control of the WiFi interface. The detection algorithm incorporated a channel hopping mechanism across channels 1–13 with a dwell time of 200–300 ms, determined experimentally as the optimal trade off between packet capture accuracy and energy efficiency. Shorter dwell times decreased detection accuracy, while longer durations increased latency and power consumption. A detection threshold was established through calibration experiments, where the presence of more than 30 deauthentication frames within 5 seconds was classified as an active attack.

The testing and validation phase was conducted through controlled deauthentication attack simulations using a WiFi Deauther device in both laboratory and public network environments. Four testing scenarios normal, low, moderate, and high attack intensities were evaluated, each repeated five times to ensure consistency and reliability of results. The primary performance indicators included detection accuracy, notification latency, and power consumption. Power usage was measured using a digital ammeter connected to the ESP32 power line, and average current readings were compared across scenarios to assess energy efficiency.

Subsequently, data analysis and evaluation were performed to assess trade-offs between detection performance and energy consumption. Detection accuracy was calculated using standard statistical metrics true positive, false positive, and false negative while average latency and current consumption were derived from repeated trials. The integration of these performance metrics enabled verification of both the system's reliability and its energy-saving capability. The methodological rationale behind this experimental design was to ensure quantitative verification and reproducibility of the results. The utilization of the ESP32 microcontroller allowed practical implementation of a low cost, portable, and energy-efficient IoT-based network security system. Furthermore, the structured testing and analysis framework ensured that the system not only achieved technical robustness but also contributed to enhancing digital security literacy through real-time user notifications.

Result and Discussion

The ESP32-based deauthentication attack detection algorithm is designed to operate in real-time by leveraging the Wi-Fi module's promiscuous mode. This algorithm captures 802.11 management frames to identify anomalous patterns, specifically deauthentication or disassociation frames, and then escalates notifications through a monitoring dashboard and a Telegram bot. The main stages of the algorithm begin with connection initialization, where the ESP32 connects to a valid Access Point (AP) to send data to a backend server via MQTT/HTTP protocols, while simultaneously activating promiscuous mode to monitor management frame traffic. Next is frame monitoring, where the module performs channel hopping across channels 1–13 at specific intervals (200–300 ms) to filter for frames with Deauthentication (0x0C) or Disassociation (0x0A) subtypes. This is followed by anomaly detection, using a sliding window counter to count these frames within a set period (5 seconds). If the frame count exceeds a predefined threshold (>30 frames in 5 seconds or >10 frames from the same source), the system classifies the event as an attack. Upon detection, a local indication is triggered, causing an RGB LED to flash red and a buzzer to sound a brief alert. Concurrently, notification to the dashboard and Telegram bot occurs, where attack data

https://doi.org/10.58291/ijec.v4i2.433

including BSSID, source address, channel, RSSI, frame count, and timestamp is sent via the MQTT protocol to a broker. The backend then stores this data, displays a visualization on the dashboard, and sends an instant notification to the user or network administrator. Before the program is written to the ESP32 control board, pseudocode is created to simplify debugging and error tracing. The pseudocode for the ESP32-Based Deauthentication Attack Detection is presented as follows.

```
=============================================
Real-Time Deauthentication Detection Algorithm
========================================

// Initialization
init_system ()
connect_to_AP (SSID, PASSWORD) //
connect to access point for internet & dashboard
setup_MQTT (BROKER_IP, TOPIC) //
initialize connection to MQTT broker
enable_promiscuous_mode(filter=MGMT) //
enable promiscuous mode only for management frames
set_hop_schedule (channels=1...13,
dwell=200ms)

while (true):
   for channel in hop_schedule:
      set_wifi_channel(channel)
      t0 = current_time ()

      // Listen during dwell time
      while current_time () - t0 < dwell:
         packet = sniff_packet ()

         if is_management_frame(packet):
            subtype = get_subtype(packet)

            if subtype == DEAUTH or subtype
== DISASSOC:
               sa   = packet.source_address
               da   = packet.destination_address
               bssid= packet.bssid
               rssi = packet.rssi
               reason= packet.reason_code

               update_counters (sa, bssid,
channel, reason, rssi)

      // Evaluate anomaly after dwell is complete
      if detect_anomaly (window=5s, threshold=30):
         // Local indication
         led_rgb (red, blink=3)
         buzzer_beep(short)

         // Prepare JSON payload
         payload = {
            "timestamp": now (),
            "bssid": bssid,
            "source": sa,
            "channel": channel,
            "reason": reason,
            "count":
get_count(bssid),
            "rssi_max":
get_rssi_max(bssid)
         }

         // Send to dashboard (MQTT)
         mqtt_publish (TOPIC, payload)

         // Send Telegram notification via
HTTP webhook
         http_post (TELEGRAM_API_URL, payload)
       // Reset or decay counters to avoid accumulation
      decay_counters ()
   sleep_short () // save energy
```

This algorithm combines two ESP32 functions, namely promiscuous mode for monitoring 802.11 frames and normal connection to access points for sending data to the server. With this approach, the system can detect attacks in real time while providing an instant notification mechanism that can increase users' digital security awareness. Additionally, the use of a sliding window-based threshold ensures that the system is not overly sensitive to false positives caused by legitimate deauthentication frames (e.g., when users normally log out of the network). Thus, the system achieves a high level of detection accuracy while conserving power consumption because channel hopping is performed on a duty cycle basis. The wiring diagram of the module is shown in Figure 3.

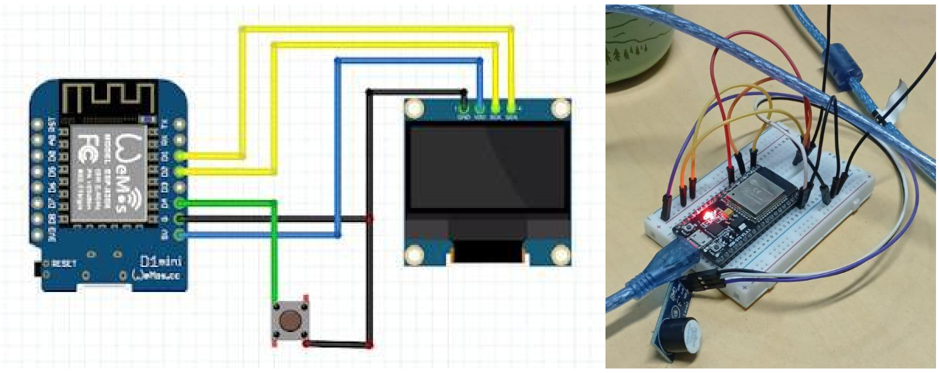


Figure 3 Wiring and prototyping of the deauthentication attack detection module

Test results show that the ESP32-based real-time deauthentication attack detection module is capable of achieving a high level of accuracy in various attack scenarios. Under normal conditions without any attacks, the system only produced one false positive case, which was most likely triggered by legitimate disassociation activity from client devices. However, the detection accuracy remained at 98.5%, so it did not cause any significant disruption to users. The test results are shown in Table 1.

Table 1 Deauthentication attack detection results

Scenario	Packet Deauth	True Positive (TP)	False Positive (FP)	False Negative (FN)	Accuration	Power Consumption (mA)	Notification Latency (ms)
Normal	0	0	1	0	98,5	68	0
Low	100	98	0	2	98	72	220
Moderate	300	296	1	4	98,3	75	250
High	600	589	2	11	97,8	79	270

In light to heavy attack scenarios, the module shows consistency with detection accuracy above 97%, although there are a few false negatives at high attack intensities. This phenomenon is thought to be due to packet collisions during the channel hopping process, where a small portion of deauthentication packets are not successfully captured. However, the false negative rate is still within the tolerance threshold, so the system can still provide relevant information in real-time, as shown in Figure 4.

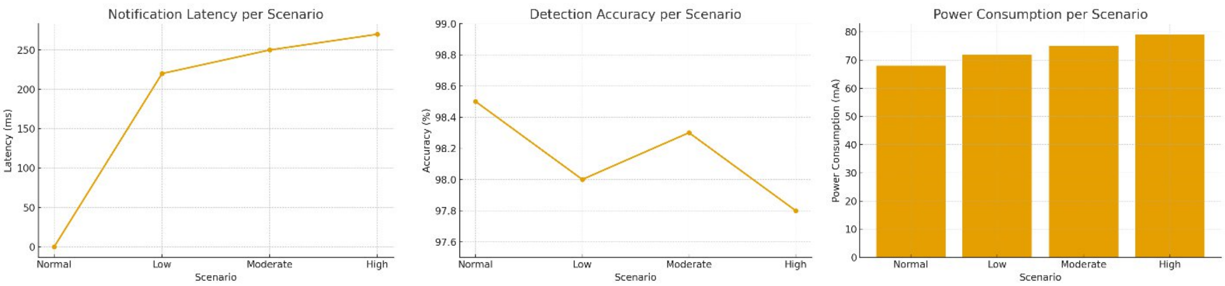
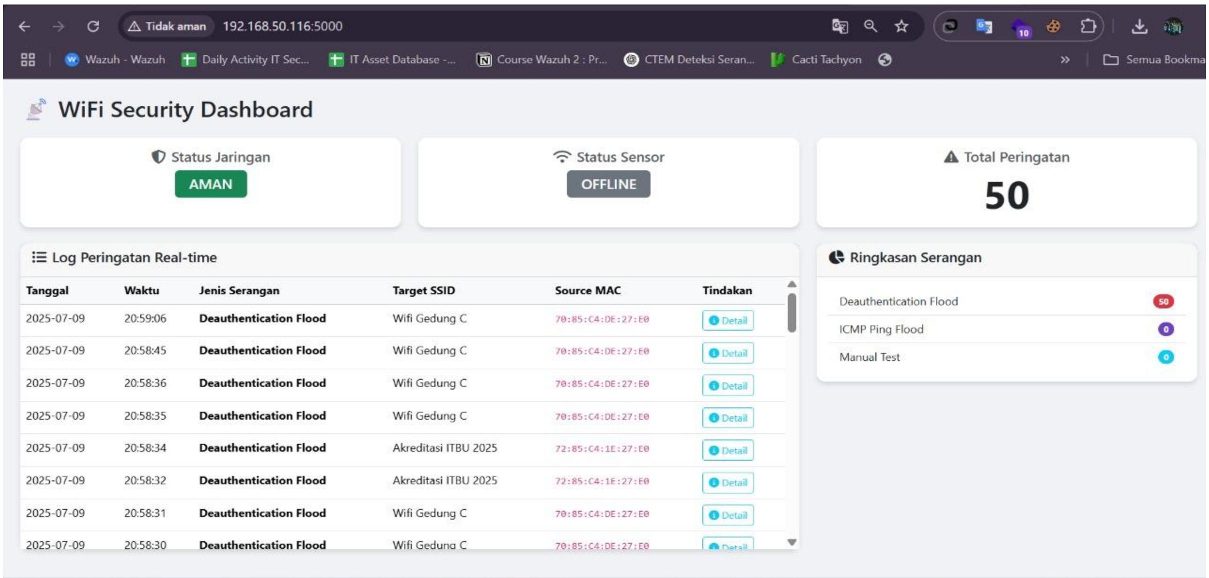


Figure 4 Accuracy, notification latency, and power consumption graphs per scenario

The application of the duty-cycle channel hopping mechanism has proven to be effective in optimizing power requirements. Average power consumption ranges from 68 to 79 mA, which is lower than the continuous monitoring mode, which generally ranges from 110 to 120 mA. This confirms that the integration of active and short sleep phases successfully reduces energy requirements without compromising detection accuracy.

The prototype module also demonstrated a fast response time in sending notifications. The average latency for sending data to the MQTT-based dashboard and Telegram bot was recorded at under 300 ms, which can still be categorized as real-time and sufficient to provide early warnings to users. The combination of visual indicators via RGB LEDs and buzzers with a monitoring dashboard and digital notifications makes this module not only effective but also easy to understand for lay users in the context of digital security literacy. The monitoring dashboard for deauthentication attack detection is shown in Figure 5. The overall evaluation results show that this module successfully fulfills the research objectives, namely providing an accurate, low-power, real-time deauthentication attack detection system. These findings reinforce the potential use of ESP32 as a low-cost and practical solution in the development of IoT devices aimed at improving public digital security literacy.



WiFi Security Dashboard

Status Jaringan: AMAN

Status Sensor: OFFLINE

Total Peringatan: 50

Log Peringatan Real-time

Tanggal	Waktu	Jenis Serangan	Target SSID	Source MAC	Tindakan
2025-07-09	20:59:06	**Deauthentication Flood**	Wifi Gedung C	70:85:C4:DE:27:E0	Detail
2025-07-09	20:58:45	**Deauthentication Flood**	Wifi Gedung C	70:85:C4:DE:27:E0	Detail
2025-07-09	20:58:36	**Deauthentication Flood**	Wifi Gedung C	70:85:C4:DE:27:E0	Detail
2025-07-09	20:58:35	**Deauthentication Flood**	Wifi Gedung C	70:85:C4:DE:27:E0	Detail
2025-07-09	20:58:34	**Deauthentication Flood**	Akreditasi ITBU 2025	72:85:C4:1E:27:E0	Detail
2025-07-09	20:58:32	**Deauthentication Flood**	Akreditasi ITBU 2025	72:85:C4:1E:27:E0	Detail
2025-07-09	20:58:31	**Deauthentication Flood**	Wifi Gedung C	70:85:C4:DE:27:E0	Detail
2025-07-09	20:58:30	**Deauthentication Flood**	Wifi Gedung C	70:85:C4:DE:27:E0	Detail

Ringkasan Serangan

Deauthentication Flood	50
ICMP Ping Flood	0
Manual Test	0

Figure 5 Deauthentication attack detection monitoring dashboard

Conclusions

This research successfully developed an energy-efficient, ESP32-based IoT module for the real-time detection of deauthentication attacks. The system achieved over 97% detection accuracy with 30–40% greater power efficiency, thanks to a duty-cycle channel-hopping mechanism. Instant notifications (<300 ms) via MQTT and Telegram make it an effective early warning tool and a medium for improving digital security literacy. The findings demonstrate that technical innovation in low-cost IoT systems can directly contribute to raising user awareness and strengthening public cybersecurity resilience. This outcome highlights the practical significance of bridging technical development with human-centered digital literacy efforts. Compared with prior studies that relied on high-specification devices or complex configurations, this research introduces a novel integration of lightweight IoT hardware and energy-efficient algorithms, expanding the accessibility of real-time attack detection solutions. Nonetheless, limitations remain, particularly the occurrence of false negatives under high-intensity attacks, which indicate the need for adaptive detection algorithms. Future studies should explore these directions further, focusing on scalability, cross-platform integration, and user-centered evaluation to maximize both technical reliability and societal impact.

Acknowledgements

The author would like to thank the Ministry of Education, Culture, Research, and Technology for its financial support through the Early Career Faculty Research Scheme, based on Decision Letter No. 0419/C3/DT.05.00/2025 and Agreement/Contract No. 0919/LL3/AL. 04/2025, 411/Rek/ITBU/VI/2025. This research also received facility support from the Budi Utomo Institute of Technology. Thanks, are also extended to the head of the DPPM Budi Utomo Institute of Technology for academic guidance, as well as to colleagues in the Informatics Engineering Study Program who assisted in the testing and data analysis process until the completion of this research.

References

Addison, S. K., Tahir, M., & Isoaho, J. (2025). Experimental Implementation of a Low Cost Real-Time Threat Intelligence Solution for Smart Home Security. *Procedia Computer Science*, *257*, 575–582. https://doi.org/10.1016/j.procs.2025.03.074

Ahadi, S. A. A., Rakesh, N., & Varshney, S. (2020). Overview on Public Wi-Fi Security Threat Evil Twin Attack Detection. *Proceedings of IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation, ICATMRI 2020*, 1–6. https://doi.org/10.1109/ICATMRI51801.2020.9398377

Allafi, R., & Alzahrani, I. R. (2024). Enhancing Cybersecurity in the Internet of Things Environment Using Artificial Orca Algorithm and Ensemble Learning Model. *IEEE Access*, *12*(March), 63282–63291. https://doi.org/10.1109/ACCESS.2024.3390093

Arreaga, N. X., Enriquez, G. M., Blanc, S., & Estrada, R. (2023). Security Vulnerability Analysis for IoT Devices Raspberry Pi using PENTEST. *Procedia Computer Science*, *224*, 223–230. https://doi.org/10.1016/j.procs.2023.09.031

Gebresilassie, S. K., Rafferty, J., Chen, L., Cui, Z., & Abu-Tair, M. (2023). Transfer and CNN-Based De-Authentication (Disassociation) DoS Attack Detection in IoT Wi-Fi Networks. *Electronics (Switzerland)*, *12*(17). https://doi.org/10.3390/electronics12173731

Harkai, A. (2024). Managing cyber-security risks associated with IoT devices for conducting financial transactions within the smart home ecosystem. *Procedia Computer Science*, *242*, 200–210. https://doi.org/10.1016/j.procs.2024.08.260

Litayem, N., & Al-Sa’di, A. (2023). Exploring the Programming Model, Security Vulnerabilities, and Usability of ESP8266 and ESP32 Platforms for IoT Development. *2023 IEEE 3rd International Conference on Computer Systems, ICCS 2023*, *November*, 150–157. https://doi.org/10.1109/ICCS59700.2023.10335558

Padhy, S., Alowaidi, M., Dash, S., Alshehri, M., Malla, P. P., Routray, S., & Alhumyani, H. (2023). AgriSecure: A Fog Computing-Based Security Framework for Agriculture 4.0 via Blockchain. *Processes*, *11*(3). https://doi.org/10.3390/pr11030757

Park, J., Kim, J., Gupta, B. B., & Park, N. (2021). Network Log-Based SSH Brute-Force Attack DetectionModel. *Computers, Materials and Continua*, *68*(1), 887–901.

https://doi.org/10.32604/cmc.2021.015172

Permana, R., Yuswanto, A., & Wibowo, B. (2023). Nodemcu Internet of Things-Based Fire Automation Design With Blynk Apps Notifications. *Teknokom*, *6*(1), 14–19. https://doi.org/10.31943/teknokom.v6i1.94

Riza, F. (2023). Analisis Security Information And Event Management (SIEM) Elastic Search Menggunakan Metode NIST 800-61 REV2 Pada Datacenter PT. Sembilan Pilar Semesta. ISMETEK, 16(2). http://ismetek.itbu.ac.id/index.php/jurnal/article/view/213

Schepers, D., Ranganathan, A., & Vanhoef, M. (2022). On the Robustness of Wi-Fi Deauthentication Countermeasures. *WiSec 2022 - Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 245–256. https://doi.org/10.1145/3507657.3528548

Selvarathinam, N. S., Dhar, A. K., & Biswas, S. (2019). Evil twin attack detection using discrete event systems in IEEE 802.11 Wi-Fi networks. *27th Mediterranean Conference on Control and Automation, MED 2019 - Proceedings*, 316–321. https://doi.org/10.1109/MED.2019.8798568

Soner, O., Kayisoglu, G., Bolat, P., & Tam, K. (2024). Risk sensitivity analysis of AIS cyber security through maritime cyber regulatory frameworks. *Applied Ocean Research*, *142*(November 2023), 103855. https://doi.org/10.1016/j.apor.2023.103855

Wibowo, B., & Yuswanto, A. (2023). The Early Detection of LPG Gas Cylinder Leaks in Households Based on the Internet of Things with SMS Message Notifications. *Jurnal Komputer Dan Elektro Sains*, *1*(1), 1–4. https://doi.org/10.58291/komets.v1i1.87

Yazdinejad, A., Zolfaghari, B., Azmoodeh, A., Dehghantanha, A., Karimipour, H., Fraser, E., Green, A. G., Russell, C., & Duncan, E. (2021). A review on security of smart farming and precision agriculture: Security aspects, attacks, threats and countermeasures. *Applied Sciences (Switzerland)*, *11*(16). https://doi.org/10.3390/app11167518

Zahid, S., Mazhar, M. S., Abbas, S. G., Hanif, Z., Hina, S., & Shah, G. A. (2023). Threat modeling in smart firefighting systems: Aligning MITRE ATT&CK matrix and NIST security controls. *Internet of Things (Netherlands)*, *22*(March), 1–18. https://doi.org/10.1016/j.iot.2023.100766

Zelle, D., Plappert, C., Rieke, R., Scheuermann, D., & Krauß, C. (2022). ThreatSurf: A method for automated Threat Surface assessment in automotive cybersecurity engineering. *Microprocessors and Microsystems*, *90*(February), 104461. https://doi.org/10.1016/j.micpro.2022.104461