Internet of Things 31 (2025) 101602

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot

Research article

Invisible eyes: Real-time activity detection through encrypted Wi-Fi traffic without machine learning

Muhammad Bilal Rasool [a], Uzair Muzamil Shah [a], Mohammad Imran [a], Daud Mustafa Minhas [b], Georg Frey [c],*

[a] *Department of Cyber Security, Air University, Islamabad, 44230, Pakistan*
[b] *Industrial Security Lab, ZeMA – Center for Mechatronics and Automation Technology, Saarbrücken, D-66121, Saarland, Germany*
[c] *Chair of Automation and Energy Systems, Saarland University, Saarbrücken, D-66123, Saarland, Germany*

ARTICLE INFO

Keywords:
Cybersecurity threats
IoT security
Network security
Lightweight attack
Privacy vulnerability
Real-time monitoring
Surveillance privacy risks

ABSTRACT

Wi-Fi camera-based home monitoring systems are increasingly popular for improving security and real-time observation. However, reliance on Wi-Fi introduces privacy vulnerabilities, as sensitive activities within monitored areas can be inferred from encrypted traffic. This paper presents a lightweight, non-ML attack model that analyzes Wi-Fi traffic metadata—such as packet size variations, serial number sequences, and transmission timings—to detect live streaming, motion detection, and person detection. Unlike machine learning-based approaches, our method requires no training data or feature extraction, making it computationally efficient and easily scalable. Empirical testing at varying distances (10 m, 20 m, and 30 m) and under different environmental conditions shows accuracy rates of up to 90% at close range and 72% at greater distances, demonstrating its robustness. Compared to existing ML-based techniques, which require extensive retraining for different camera manufacturers, our approach provides a universal and adaptable attack model. This research underscores significant privacy risks in Wi-Fi surveillance systems and emphasizes the urgent need for stronger encryption mechanisms and obfuscation techniques to mitigate unauthorized activity inference.

1. Introduction

The Wireless camera technology has revolutionized the field of surveillance and monitoring, offering unprecedented flexibility, convenience, and accessibility [1]. In contrast to conventional wired systems, wireless cameras function via Wi-Fi networks, removing the requirement for intricate wiring setups and allowing remote access from any location with an Internet connection [2]. Numerous industries, including retail, public safety, industrial monitoring, and home security, have widely adopted this technology [3].

Wireless camera security is crucial for protecting people and property in today's connected world. These cameras provide enterprises, government organizations, and homes with real-time surveillance capabilities, enabling rapid responses to security incidents and proactive threat detection [4]. With advanced features like motion detection, night vision, and remote mobile access [5,6], wireless cameras offer users convenient, continuous monitoring of their surroundings. In business environments, they are essential for safeguarding assets, preventing theft and damage, and ensuring compliance with industry regulations [7].

Wireless cameras with features like motion detection, night vision, and two-way voice communication offer consumers flexible monitoring options tailored to their needs [8]. However, transmitting video data over Wi-Fi networks poses inherent privacy and

* Corresponding author.
E-mail address: georg.frey@aut.uni-saarland.de (G. Frey).

https://doi.org/10.1016/j.iot.2025.101602
Received 10 January 2025; Received in revised form 9 March 2025; Accepted 29 March 2025
Available online 11 April 2025

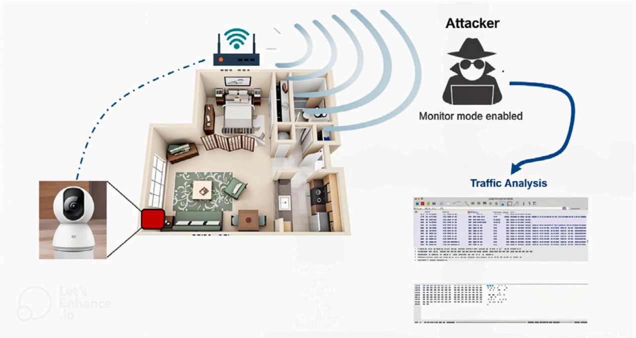


Fig. 1. A typical scenario for home surveillance attack.

security risks, including interception and unauthorized access. As the use of wireless cameras expands, addressing these privacy and security challenges is essential to ensure the integrity of surveillance systems in the digital age [9]. The widespread use of cameras and internet-connected devices has increased the vulnerability of sensitive data to unauthorized access [10]. To prevent malicious interception and misuse of video data, robust security measures such as encryption, secure authentication protocols, and regular firmware updates are essential. Prioritizing wireless camera security enables users to mitigate potential risks and fully benefit from surveillance technology without compromising privacy or security [11]. This introduction sets the stage for exploring wireless camera technology, its applications, and the evolving landscape of privacy and security in the surveillance industry [12].

Although wireless cameras offer convenience and flexibility, they are vulnerable to security flaws that threaten user privacy and data integrity. A significant risk is the potential for attackers to capture and analyze Wi-Fi traffic, which allows them to infer critical information about activities within the camera's field of view, as shown in Fig. 1 [13,14]. This misuse of encrypted Wi-Fi traffic endangers user privacy, enabling unauthorized access to potentially sensitive data without user consent [15]. Wireless cameras are also susceptible to cyberattacks, such as hacking or unauthorized access to their firmware and control panels. Cybercriminals can exploit unsecured cameras as entry points into a user's network, risking other connected devices and sensitive data [16]. Vulnerabilities in camera software or firmware can allow attackers to bypass security measures, alter settings, or even take remote control of the device [17]. Given the growing use of wireless cameras, addressing these vulnerabilities with robust security measures is essential to prevent unauthorized access and maintain the integrity of surveillance systems.

While this study focuses on privacy risks in Wi-Fi-based surveillance cameras, similar vulnerabilities exist across various IoT devices, including smart speakers, doorbell cameras, and home automation systems. Prior research has shown that encrypted traffic from smart assistants (e.g., Amazon Echo, Google Home) can reveal user interactions through statistical traffic analysis. Likewise, motion sensors, smart locks, and other security devices generate unique traffic patterns that attackers can exploit to infer user behavior. Compared to these devices, Wi-Fi cameras pose an even greater risk, as they continuously transmit data and generate distinct metadata patterns that are easier to analyze. Understanding these broader vulnerabilities emphasizes the need for holistic security enhancements in IoT ecosystems, not just in surveillance systems.

Existing research in encrypted Wi-Fi traffic analysis primarily relies on machine learning-based methods to infer activities. Techniques such as deep learning, statistical classifiers, and pattern recognition models have been widely used to detect IoT device activities from encrypted traffic. While these methods can achieve high accuracy, they require extensive labeled training datasets, high computational resources, and frequent retraining to generalize across different environments. However, an ML model trained for a specific camera manufacturer may not yield the same results for cameras from different manufacturers, meaning an attacker would need to train multiple models for various devices to maintain accuracy. In contrast, our proposed method introduces a lightweight, non-ML-based attack model that infers activities solely from traffic metadata and packet behavior. Unlike ML-based models, our approach remains effective without requiring feature extraction, data labeling, or iterative learning, making it more scalable and computationally efficient for an attacker. Instead of requiring multiple ML models for different camera brands, our approach simply analyzes recurring traffic patterns to detect live streaming, motion detection, and person detection—allowing an attacker to deploy a simple program for immediate activity inference. This research fills a significant gap by demonstrating that even without sophisticated AI models, encrypted traffic patterns can reveal sensitive surveillance activities, underscoring the need for stronger privacy protections in Wi-Fi monitoring systems.

1.1. Related work

This study aims to enhance understanding of the diverse strategies employed by researchers to extract sensitive data from encrypted traffic, particularly in relation to Wi-Fi surveillance devices. A comparative analysis of previous literature is shown in Table 1.

The researchers in [18] began by capturing Wi-Fi traffic data from multiple surveillance systems and preprocessed the data by removing *I* reference frames used in video compression. They segmented the traffic data into time intervals (X) and computed the rate of traffic change (C) within each segment to capture data flow variations. Statistical features, including mean, variance,

Table 1
Comparative analysis of prior research on IoT traffic analysis and privacy risks.

Author(s)	Research objective	Methodology	Limitations
Liu, Xiaolong, et al. [18]	To infer user behaviors from encrypted video surveillance traffic using machine learning techniques.	Employed machine learning models to analyze patterns in encrypted video surveillance traffic and infer activities. Used features like packet size and transmission timing to identify distinct behaviors.	Relies heavily on machine learning accuracy, which can be affected by changes in network conditions.
Apthorpe, Noah, et al. [19]	To examine privacy risks and defenses related to encrypted IoT traffic in smart homes.	Conducted traffic pattern analysis to identify IoT activities, then proposed countermeasures for data privacy. Used statistical analysis to map traffic bursts to user activities.	High computational costs for encryption, and defensive measures are limited by variable IoT encryption schemes.
Huang, Qianyi, et al. [20]	To evaluate privacy risks posed by wireless surveillance cameras in smart homes.	Analyzed packet size and frequency in encrypted traffic from surveillance cameras to identify activities. Examined different scenarios for potential information leakage.	Limited generalizability across various camera models and network setups, as behavior may vary.
M. Alyami, I. Alharbi, et al. [21]	To develop a profiling attack on IoT devices over Wi-Fi using eavesdropping.	Profiled devices by analyzing packet length and timing intervals in Wi-Fi traffic to detect device types and usage. Applied statistical classification to profile devices based on traffic signatures.	Only effective in Wi-Fi networks and requires constant monitoring for optimal classification accuracy.
Pinheiro, A.J., et al. [22]	To identify IoT devices and events by analyzing encrypted traffic packet lengths.	Applied traffic profiling to recognize devices based on consistent packet length patterns and detect certain events. Implemented statistical correlation between packet lengths and device activities.	Limited adaptability to changes in encryption patterns, and results vary based on packet consistency.
Acar, Abbas, et al. [23]	To reveal smart home activities through encrypted network traffic analysis.	Mapped distinct traffic characteristics to user activities in smart homes. Used packet analysis to detect patterns related to specific activities like motion and video streaming.	Model limited to specific devices; decreases in accuracy with increasing diversity of IoT devices.
Dong, Shuaike, et al. [24]	To fingerprint IoT devices in smart home environments and infer activities.	Used automated traffic profiling techniques to identify IoT devices by observing unique packet length and timing. Applied fingerprinting to infer user activity based on network traffic patterns.	Limited to device-specific patterns, requiring updates for new device types and protocols.
Edu, Jide S., et al. [25]	To review security and privacy vulnerabilities in smart home personal assistants.	Conducted a systematic literature review to identify major privacy and security risks in smart home assistants. Examined current protective measures and highlighted gaps in personal assistant security.	Mostly theoretical; lacks empirical testing across different smart home devices and configurations.
Mari, Daniele, et al. [26]	To infer indoor scenes from encrypted video-surveillance traffic by analyzing traffic patterns.	Applied encrypted traffic pattern analysis to deduce room activities and scene changes. Focused on packet size and frequency to approximate real-world actions.	Varies in effectiveness with encryption variability and requires high data volume for scene inference accuracy.

skewness, and kurtosis, were calculated for both X and C to capture key traffic pattern characteristics, such as central tendency, dispersion, and asymmetry. These features were then analyzed to identify patterns and anomalies in the Wi-Fi traffic, supporting the inference of behaviors from encrypted surveillance data.

While, in study [19], researchers investigated privacy attacks and defenses in encrypted IoT traffic within smart home environments. They collected a dataset of encrypted communications from various smart devices, including assistants, cameras, and thermostats, and conducted preprocessing to clean the data. Using advanced traffic analysis techniques, they identified potential privacy vulnerabilities by deriving insights about user behavior, device interactions, and environmental variables from patterns, anomalies, and correlations. The study also evaluated defense strategies aimed at mitigating privacy risks without hindering device functionality, providing an in-depth analysis of privacy threats and protective measures in smart home IoT networks.

Moreover, the researchers in [20] used a methodical approach to evaluate privacy risks related to wireless surveillance cameras in the study. Three different types of packets were identified by the researchers after they gathered the encrypted video stream data: Type A, which stood for packets with maximum length units (e.g., 1,500 bytes); Type B, which stood for packets with fixed sizes but less than 1,500 bytes; and Type C, which stood for packets with variable sizes.

In addition, [21] highlights the ease of profiling IoT devices from outside a Wi-Fi network. Researchers captured bidirectional traffic flows associated with the Wi-Fi router's MAC address, filtering out noise frames and pairing device names and statuses with MAC addresses to enhance profiling accuracy. They developed a Python script to extract and analyze statistical features indicative of device behavior. Using machine learning models such as Random Forest, Support Vector Machine, and Naïve Bayes, the researchers identified patterns and inferred device identities. The study demonstrates the effectiveness of eavesdropping on encrypted Wi-Fi traffic for device profiling, emphasizing the need for stronger security measures to counteract such vulnerabilities.

In study [22], researchers proposed a method based on packet length statistics to detect IoT devices and events while distinguishing them from non-IoT sources. Key metrics included total bytes transferred per second, mean packet length, and standard deviation. Traffic data from each device was segmented into one-second intervals for easier analysis. Using a dataset of 748,443 samples, the researchers applied algorithms such as K-Nearest Neighbors, Decision Tree, Random Forest, Support Vector Machine, and Majority Voting to assess each metric's contribution to device identification. Their findings indicated that mean packet length was the most significant factor. This packet length-based analysis effectively identified IoT devices and events from encrypted data, highlighting its potential for exploiting IoT security and privacy in complex network environments.

In [23], the authors analyzed encrypted traffic patterns to assess smart home privacy vulnerabilities, proposing a method to detect device states and infer user activities through sequence analysis of packet lengths and timing. Similarly, [24] focused on automated fingerprinting of IoT traffic to identify smart home devices, using packet lengths and inter-packet intervals to characterize device behaviors accurately. Both studies underscore the privacy risks associated with encrypted IoT traffic and the need for effective strategies to protect smart home privacy.

Whereas, [25] provides a comprehensive examination of the security and privacy implications associated with smart home personal assistants. The authors explore various vulnerabilities and risks inherent in these devices, such as unauthorized access to sensitive information and potential privacy breaches. They highlight the importance of implementing robust security measures to mitigate these risks and protect user privacy. Likewise, the feasibility of inferring scenes from video-surveillance encrypted traffic is done in [26]. By analyzing encrypted traffic patterns, the authors demonstrate the potential for unauthorized parties to gain insights into video surveillance activities, posing significant privacy concerns.

1.2. Our contribution

The article's primary contributions focus on introducing a lightweight, non-ML-based approach to infer activities from encrypted Wi-Fi traffic associated with wireless cameras. It presents a unique attack model that deviates from the trend of inferring granular user activities using complex techniques. Moreover, it explicitly pinpoints a vulnerability related to the notification mechanisms commonly found in Wi-Fi cameras. While these notifications may be encrypted, their presence, timing, and accompanying traffic patterns inadvertently expose critical information about the monitored environment. This specific vulnerability has not been heavily explored in existing research, highlighting a previously underestimated privacy risk associated with seemingly secure features. Eventually, the research goes beyond identifying the vulnerability and explores potential mitigation strategies. Some of the highlighted contributions are:

- This paper presents a practical attack model that allows for inferring specific camera activities—such as live streaming, motion detection, and person detection—using only encrypted Wi-Fi traffic patterns. This approach bypasses the need for complex machine learning models, offering a new perspective on activity detection without substantial computational resources.
- The paper demonstrates how variations in packet size, serial number patterns, and packet arrival times can reliably indicate distinct activities.
- The vulnerability mitigation approach is demonstrated to be scalable, working effectively in networks with multiple devices while maintaining high accuracy, particularly at shorter distances. The simplicity of the proposed model allows for real-time monitoring without relying on extensive datasets or frequent updates required for ML-based techniques.
- Through experimentation at varying distances (10 m, 20 m, and 30 m), the model's performance is validated, with a notable decline in accuracy as signal strength weakens over greater distances. This offers insights into the model's robustness in real-world conditions and highlights the trade-offs in packet loss and signal strength on detection accuracy

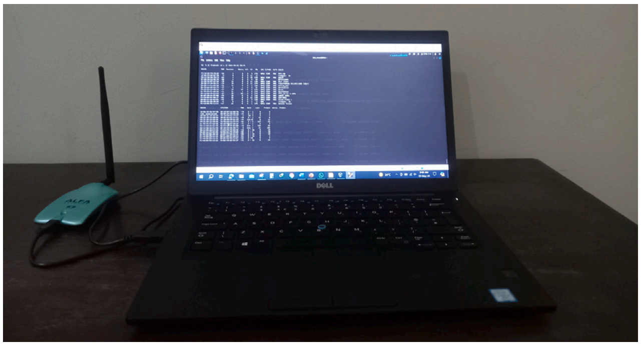


Fig. 2. Setup for capturing Wi-Fi traffic.

- The paper underscores the privacy risks inherent in Wi-Fi-based camera systems by illustrating how even encrypted traffic can reveal sensitive information. These insights are crucial for informing manufacturers and policymakers on the need for enhanced privacy protections in Wi-Fi surveillance systems

The rest of the paper is organized as follows. In Section 2, an attack model and experimental setup are proposed. Section 3 presents activity detection and inference techniques to predict the actions from Wi-Fi traffic. The results and discussions are presented in Section 4 followed by discussing the effectiveness and scalability of the attack model, mitigation techniques, and their comparison to the existing techniques. Finally, the paper is concluded in Section 5.

2. Proposed attack scenario

2.1. Attack scenario and model assumptions

Attack scenario centers on an intruder who plans to break into a home or building while its occupants are away and the owner is not actively watching the live feed from the camera. Theft, vandalism, or other malicious behaviors could be the attacker's motivations. The attack is especially worrying because of how easy it is to set up, attacker needs a device with Wi-Fi monitoring capabilities and a program developed to record and examine Wi-Fi traffic. The potential for widespread exploitation of Wi-Fi camera systems is increased by this low barrier to entry.

The attacker places itself in the target Wi-Fi network's range to carry out the attack. It uses a Wi-Fi adaptor that has the ability to operate in monitor mode to intercept any wireless traffic that is sent over the network. The attacker can intercept data packets having details about network activities using this mode. The attacker can filter and analyze this traffic to deduce important information about the condition of the monitored environment by executing specific program. The attack takes advantage of the fact that patterns and metadata suggestive of particular behaviors can be found in even encrypted transmission.

The main component of the attack is to determine whether the homeowner is actively viewing the live video stream and if someone is present in the house. The traffic patterns of notifications sent by security cameras allows for these inferences. Many contemporary home security systems incorporate person and motion detection capabilities that send alerts to the owner's mobile. These notifications are encrypted, but their existence and timing can be examined to uncover hidden activity. Through close inspection of the traffic that has been recorded, the attacker can pinpoint times when there is movement or human activity as well as times when the owner is watching the camera stream in real time.

For the attack model to function effectively, it is essential that notifications are enabled on the home surveillance system. These notifications generate consistent traffic patterns, which an attacker could potentially detect and analyze. This situation reveals a critical vulnerability in Wi-Fi camera systems, where sensitive information about the household's status can be inferred without decrypting the actual notification content. This highlights the importance of understanding and mitigating the security risks associated with Wi-Fi surveillance systems.

The attack model is not confined to the above scenario but could extend to the following schemes:

- Indirect Pattern Detection: Even if users restrict push notifications on their mobile devices, the surveillance app may still receive updates internally. In such cases, the camera continues to send activity updates to the app, though they do not appear on the user's mobile screen. By analyzing the network traffic generated by the app, the model could infer the timing of these notifications, even when they remain unseen by the user.
- Patterns in Activity: A limited number of notifications can reveal patterns indicative of user behavior. For instance, if notifications occur predominantly at specific times (e.g., early morning or late evening), this could indicate the hours when the cameras are actively used, offering insights into the household's daily routines and monitoring habits.
- Establishing Baseline Data: Over time, a baseline of the timing and frequency of "normal" notification can be established. This baseline enables the model to differentiate between periods when the system is actively monitored versus ignored or disabled, offering indirect insights into user behavior and the camera's operational status.

- Unexpected Spikes in Notifications: Notifications occurring at irregular or unexpected times could signal specific events. For example, a spike in notifications outside usual hours might suggest a sudden or unusual activity, such as the user checking in unexpectedly or responding to an event that requires attention (e.g., movement by a pet or an intruder).
- Absence of Activity as a Signal: The absence of notifications during certain times can also be informative. For example, if a camera typically generates notifications but stops doing so for extended periods, this may indicate when the user is home and has disabled notifications, or when monitoring is temporarily paused. Such gaps can reveal periods of reduced user vigilance or activity, even if explicit alerts are limited.

2.2. Setup configuration

The experimental setup involves establishing a wireless network to which multiple devices, including a Wi-Fi-enabled surveillance camera, are connected. Additionally, a Wi-Fi adaptor capable of recording network traffic in monitor mode is utilized, as depicted in Fig. 2. This setup allows the simulation of a real-world scenario, where an attacker could potentially infer sensitive information by intercepting and analyzing encrypted Wi-Fi traffic. The configuration and components of this setup are crucial for the accurate replication of the attack scenario and the validation of the results.

2.2.1. Wireless network

To replicate a real-world scenario, a wireless network is established, featuring multiple devices commonly found in residential or office environments. This network is made up of numerous devices that were linked to a central Wi-Fi access point, including laptops, mobile phones, smart home appliances, and other IOT devices. The objective is to replicate the complexity and diversity of a typical Wi-Fi network by integrating a variety of devices. The network uses the 2.4 GHz frequency band, which is popular for Wi-Fi communication because it works well with a variety of devices and can pass through obstructions.

2.2.2. Wireless camera

Wi-Fi-enabled security cameras from a leading commercial brand known for their specific features were chosen to meet the research requirements [27]. Because of their robust features, affordability, and ease of use, these cameras are well-liked by customers and are frequently found in both residential and commercial settings. For traffic analysis, two distinct models of these cameras are utilized: the *360° Camera (1080p)* and the *Home Security Camera (1080p)* with a magnetic mount. These two cameras have the capacity to record video in 1080p resolution. They also make use of the cutting-edge *h.265* video codec, which effectively compresses and transmits video data.

2.2.3. Operating system

To conduct the research in a controlled and isolated setting, *VM VirtualBox 7.0* is employed to establish a virtualized environment. VirtualBox is a potent *x86* and *AMD64/Intel64* virtualization solution. It enables efficient hardware resource sharing by enabling the operation of numerous virtual machines (VMs) on a single physical machine. With this configuration, it was possible to change the environment's settings without affecting the underlying physical system. To ensure smooth operation and sufficient processing power for the tasks, the virtual machine (VM) was allocated 4 GB of RAM and 2 processors.

Kali Linux 2023.1 is installed in the VirtualBox environment as the preferred operating system. Kali Linux is a Linux distribution based on Debian that is intended for use in penetration testing and digital forensics. Within the cybersecurity community, it is well-known for its extensive pre-installed toolkit, which comes in handy for a variety of network analysis tasks [28]. The installation of Kali Linux on VirtualBox is straightforward due to its compatibility, enabling the establishment of a reliable and secure environment for research purposes.

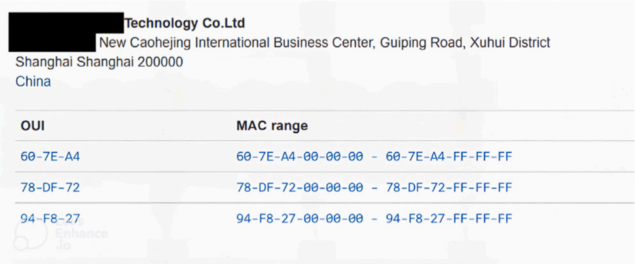
The extensive array of integrated tools for network analysis and security testing in Kali Linux proved particularly beneficial for the research. The functionalities provided by tools such as Wireshark, Aircrack-ng, and Kismet allowed us to collect, analyze, and interpret Wi-Fi traffic effectively. Employing these tools, a thorough analysis of the encrypted traffic patterns produced by the cameras is conducted.

2.2.4. Wi-Fi adaptor

The *Alfa AWUS036NH* wireless adaptor, known for its robust features and extended range, is utilized to capture Wi-Fi traffic effectively. This adaptor's high performance and dependability [29] makes it ideal for network analysis and security testing in affordable price. The *Alfa AWUS036NH's* compatibility with multiple operating systems, including Kali Linux, is one of its best features.

The high-gain antenna of the *Alfa AWUS036NH* greatly improves its ability to pick up Wi-Fi signals over a large area. Because it enables us to track WiFi traffic from various devices at different distances from the access point. The IEEE *802.11b/g/n* standards are supported by the adaptor, giving users flexibility in connecting to various network types and guaranteeing thorough traffic capture.

The *Alfa AWUS036NH's* ability to operate in monitor mode is crucial for the study, as it allows the adaptor to capture all nearby packets, rather than just those intended for the device. A substantial dataset of Wi-Fi traffic was collected from the cameras and various other networked devices, encompassing both encrypted and unencrypted packets. The *Alfa AWUS036NH* played a vital role in the experimental setup due to its strong reception capabilities and compatibility with Kali Linux tools, ensuring accurate and effective data collection for analysis.



[redacted] Technology Co.Ltd
New Caohejing International Business Center, Guiping Road, Xuhui District
Shanghai Shanghai 200000
China

OUI	MAC range
60-7E-A4	60-7E-A4-00-00-00 - 60-7E-A4-FF-FF-FF
78-DF-72	78-DF-72-00-00-00 - 78-DF-72-FF-FF-FF
94-F8-27	94-F8-27-00-00-00 - 94-F8-27-FF-FF-FF

Fig. 3. Finding manufacturer using first 3 bytes of MAC address.

3. Activity detection and inference techniques

For the successful implementation of the system, several key components must work together to record, analyze, and predict actions from Wi-Fi traffic. After establishing a comprehensive network environment, a significant volume of Wi-Fi traffic data is collected, including encrypted packets, utilizing the *Alfa AWUS036NH* wireless adaptor operating in monitor mode. The built-in tools of Kali Linux are utilized to process this data within a virtual machine, providing a robust and reliable platform for analysis. The primary goal of this analysis is to identify specific patterns and metadata in the recorded traffic that corresponded to various activities, such as live streaming events from the cameras, motion detection, and person detection.

3.1. Activity detection

The *airodump-ng* tool is utilized to capture the Wi-Fi traffic within the network [30]. This robust tool allowed for the monitoring and recording of activities from all devices connected to the network, including the cameras. Specifically, the focus is on collecting data during key events captured by the cameras, such as live streaming, motion detection, and person detection. Every event is meticulously recorded to ensure the accuracy and reliability of the dataset. The objective in gathering this data is to identify distinct patterns and features in the Wi-Fi traffic that could indicate various types of camera activity.

For each type of activity, more than *100* instances are collected to ensure a comprehensive analysis. This extensive dataset provides a robust foundation for the analysis, enabling the identification and understanding of the subtle differences and recurring patterns associated with each event. The gathered information contained a variety of characteristics, including encrypted traffic patterns, traffic rates, and packet sizes, all of which are essential for differentiating between distinct activities. Through meticulous analysis of these cases, the aim is to develop a methodology capable of accurately inferring camera activity solely from encrypted Wi-Fi traffic, thereby underscoring the privacy risks and vulnerabilities inherent in such systems.

3.2. Traffic analysis

In the traffic analysis phase, the "*.cap*" files containing the captured data are initially opened. Utilizing the network protocol analyzer Wireshark, the team examine the precise packet structure of the Wi-Fi traffic [31]. This examination reveals various components and characteristics of the Wi-Fi packets, including headers, payloads, and details related to encrypted traffic.

After familiarization with the packet structure, a thorough examination of the Wi-Fi packets in the captured file is conducted for further analysis. The primary objective is to detect distinct patterns and anomalies associated with various devices connected to the network and their respective activities. To identify the packets relevancy to each type of activity, it is necessary to filter and classify them based on their attributes. The goal is to derive useful insights and develop a technique for classifying camera traffic from encrypted Wi-Fi traffic through a systematic examination of these packets.

3.2.1. Camera traffic classification

For camera traffic classification, the objective is to differentiate the wireless camera traffic from the overall network traffic. This process involves identifying unique patterns and characteristics associated with the camera's data packets. Several methods are employed for device identification, depending on the information available to the attacker. By employing these methods, accurate isolation of camera traffic can be achieved, which is crucial for subsequent analysis and activity inference.

3.2.2. MAC addresses

For an attacker with access to smart home LAN data, classifying camera traffic using MAC addresses could be an essential step [19]. The Organizational Unique Identifier (OUI), which is the first three bytes of a device's MAC address, can be used to determine the device's manufacturer. The space of potential device identities can be significantly narrowed by assigning manufacturer labels to each device based on the OUI. This makes it easier to use traffic rate technique for additional classification. For example, determining that a specific traffic originates from camera's manufacturer can make it easier to separate its traffic from that of other networked devices.

For example, *hwaddress.com* and other resources can be used to identify the OUIs of cameras' manufacturer. This data, as shown in Fig. 3, aids in identifying the traffic coming from cameras among the different network-connected devices. For an observer, methods such as traffic rate monitoring are still useful even in the absence of information to the MAC addresses.

3.2.3. Traffic pattern

Finger printing devices using traffic rates is still a feasible option if MAC addresses are not enough to identify the devices in a network. It is surprising how well devices in the same category of smart home appliances can be distinguished by simple traffic features. For instance, the mean traffic volumes of two different security cameras can differ significantly, almost by an order of magnitude, over the data collection period [19]. These variations demonstrate how traffic rates by themselves can act as a differentiator, enabling the identification of particular devices without depending only on MAC addresses. This approach takes advantage of the inherent differences in data transmission patterns that are observable in a consistent manner across various devices and usage scenarios.

In more complex scenarios, additional features of network traffic can enhance device identification accuracy. These features include the proportion of *SYN* and *ACK* packets per flow, the total number of packets in a flow, and the distributions of inter-packet intervals. Each of these metrics provides unique insights into the behavior of networked devices, allowing for more precise identification. For example, the *SYN* and *ACK* packet proportions reflect the nature of device communications and their interaction with the network. Similarly, the number of packets in a flow and inter-packet interval distributions offer granular details about the frequency and pattern of data transmission. By analyzing these diverse traffic characteristics, it becomes feasible to construct detailed profiles for each device, thereby improving the accuracy and reliability of device identification in smart home environments.

In the traffic of commercial wireless cameras, three distinct types of packets can be identified: Type A, Type B, and Type C [20]. Type A packets are the maximum length units, typically around 1,500 bytes, representing the largest possible packet size that can be transmitted. Type B packets are of a fixed size but smaller than Type A packets, and Type C packets have unfixed sizes. The sequence in which these packets appear is notable: a series of Type A packets often appear consecutively, followed by a Type C packet, while Type B packets appear at a regular frequency, approximately 30 Hz. This pattern is likely due to the packetization of video frames, where the large size of video frames necessitates their division across multiple packets. Consequently, a Group of Pictures (GOP) spans several Type A packets, each carrying a full payload, and ends with a Type C packet, which contains the remaining part of the frame.

Type B packets serve a different role in maintaining the connection and transmitting the camera's status information at a predefined interval. This periodic transmission ensures continuous communication and monitoring, which is crucial for the functionality of surveillance systems. This packet transmission pattern is not unique to a single camera model; similar patterns have been observed in other commercial wireless cameras. Although the sizes of Type A and Type B packets may vary among different camera models, the overall transfer patterns exhibit significant similarities. These common features across various commercial cameras suggest a standardized approach to packet transmission in wireless surveillance systems, highlighting the potential for identifying and analyzing traffic patterns to infer device activities and status.

The detection of wireless cameras in the network traffic is achieved by utilizing MAC addresses and traffic patterns alongside the wireless camera traffic classification technique. Utilizing the MAC address associated with the manufacturer, enables the filtering of their devices from the Wi-Fi traffic. This initial filtering significantly narrow down the pool of potential devices. The unique traffic patterns which are distinguished by particular packet types and sequences, are then utilized to identify the wireless camera. The observations indicate that wireless cameras can be identified with high accuracy without relying on complex machine learning models, as the traffic patterns exhibit remarkable uniqueness and consistency. This shows that even sophisticated traffic analysis can be an effective method for identifying wireless cameras in a network.

3.3. Activity inference

After successfully classifying wireless camera traffic, the focus now shifts to identifying the activities associated with these cameras. The primary objective includes three key activities; person detection, motion detection, and live stream detection. These activities are vital for security and surveillance, each offering unique insights into the monitored environment. By meticulously analyzing these activities, the aim is to enhance the understanding of events occurring within the observed space.

A thorough analysis of the patterns in the notification traffic of Wi-Fi cameras is conducted to categorize activities related to motion detection and person detection. Through a comprehensive analysis, a distinct pattern is identified, characterized by data packet configurations unique to the notifications generated by wireless cameras. Drawing from this understanding, a robust classification pattern is developed by identifying three primary features within the Wi-Fi packet structure. The first feature is packet length, it offered valuable insights into the size and content of the data packets transmitted during motion detection and person detection events. By examining the differences in packet length between notification types, distinct patterns corresponding to specific activities were identified.

The serial number assigned to each packet serve as the second feature utilized in the classification pattern. The packets' sequential identifiers, or serial numbers allow us to monitor the transmission order and identify any irregularities or breaks from the expected sequence. The ability to differentiate between motion detection and person detection events is made possible by the sequential analysis, since each activity's associated notifications displays a unique serial number pattern according to the packet length. Thirdly, the packet arrival time serves as a crucial element in the classification process. By time-stamping the arrival of each packet, temporal relationships and patterns indicative of motion detection or person detection activities could be established. By incorporating these three characteristics into the classification model, the research is able to differentiate between various types of notifications and accurately determine the activities occurring in the observed environment.

When motion detection or person detection event occurs, camera sends notification to the mobile phone of the user. This transmission by camera does not only include notification but also a video clip of 9 s. Upon examining the camera's notification transmission, a consistent pattern is observed, wherein between 30 and 33 packets are transmitted prior to a stream of Maximum Transmission Unit (MTU) packets. After this first transmission, the next packets are mostly MTU packets with two different patterns of 30 to 33 packets each in between. An additional pattern of 30 to 33 packets signaled the end of the notification transmission sequence at the end of the MTU packet stream.

Once a clear understanding of the notification transmission pattern is established, the focus is shifted towards classifying each notification based on its type. The ability to distinguish differences among the 30 to 33 packet patterns was necessary for this classification. Through a comprehensive analysis, distinct variations in packet sizes and arrival times are identified, serving as reliable indicators of notification types. By leveraging these observable differences in packet sizes, a robust classification system is developed, effectively distinguishing between notifications for motion detection and person detection.

In the investigation of detecting live streaming events transmitted by the wireless camera, distinctive transmission patterns are uncovered that differentiate them from motion or person detection notifications. Live streaming events have a distinct transmission profile distinguished by an initial burst of MTU packets at the beginning of the transmission, in contrast to the sequential patterns seen in motion or person detection events. After this first spike, later packets show a mix of constant and variable sizes, which is different from the regular packet sizes usually linked to motion or person detection alerts. The transmission pattern of live streaming displays variability in packet sizes, serving as a distinctive feature that enables the system to differentiate live streaming events from other notifications. The detection algorithm leverages these unique transmission characteristics to accurately identify and classify live streaming events in real-time.

The proposed system incorporates basic fault detection mechanisms to ensure reliability in real-time monitoring. During traffic analysis, inconsistencies in packet sequences (e.g., missing serial numbers or irregular inter-packet intervals) trigger a validation subroutine. For instance, if a live streaming event is detected but lacks the expected MTU packet burst, the system flags this as a potential misclassification and re-analyzes the preceding 5-second window to resolve ambiguities. Similarly, sudden drops in traffic volume (e.g., due to packet loss) activate a signal strength check, allowing the system to discard low-confidence classifications. While these mechanisms mitigate errors, they are limited by environmental factors like intermittent interference or abrupt signal degradation, which may still lead to undetected faults.

3.4. Real-time monitoring system

The foundation of the research technique is a program developed with *Python 3.11.2*, which facilitates the analysis and classification of encrypted Wi-Fi data and infers activities associated with the use of wireless cameras. Fig. 4 provides a visual overview of the program's main features and components and illustrates the workflow schematically. The program methodically parses, interprets, and categorizes the Wi-Fi traffic using meticulously developed algorithm, makes it possible to identify camera activities. The program features a robust and efficient design, embodying the essence of the research while providing an effective means to examine security vulnerabilities and privacy concerns associated with wireless camera systems.

3.4.1. Program initialization

The design of essential features required for capturing and processing Wi-Fi communication packets was initiated. The capabilities of the Scapy [32] and Pyshark [33] libraries were utilized to capture and analyze data from the network in real time. This provided the foundation for further analysis and classification activities.

3.4.2. Capturing Wi-Fi networks

In the second step of the algorithm, the "*sniff*" function from the *Scapy* library, a powerful tool for packet manipulation and analysis in Python, is employed to initiate the capturing of nearby Wi-Fi networks. This function allows for effectively monitoring network traffic by capturing packets transmitted over the airwaves through a designated network interface. The program initiates the sniffing process and waits 30 seconds to actively capture data packets sent by neighboring access points. The application reads the Service Set Identifier (SSID) and the Basic Service Set Identifier (BSSID) from the packet headers as each packet is intercepted. The BSSID acts as a unique identification for the access point delivering the packet, while the SSID is a human-readable name that identifies the Wi-Fi network. These identifiers are essential for differentiating across nearby Wi-Fi networks.

The program builds a complete inventory of all detected networks by systematically storing the extracted SSIDs and BSSIDs over the 30-second capture period. By collecting information from various networks that may serve as potential targets, this data collection process ensures a comprehensive understanding of the local wireless environment.

3.4.3. Target network selection

The list of Wi-Fi networks that are recorded in the previous stage is displayed by the program when it moves into a user-interactive phase. With each network shows alongside its matching SSID, clear and comprehensive information about the nearby networks is provided. The user sees every network that has been detected on display and decide which network to target with knowledge. The target network is chosen by the user by providing the index linked to the network of their choice from the list that is shown. The process is streamlined by this index-based selection mechanism, which makes it simple and effective for the user to indicate his decision. The program adapts its operations to the user's particular investigative needs by enabling this interactive selection, which guarantees that it accurately focuses on the user-specified target network for subsequent live traffic capture and analysis.

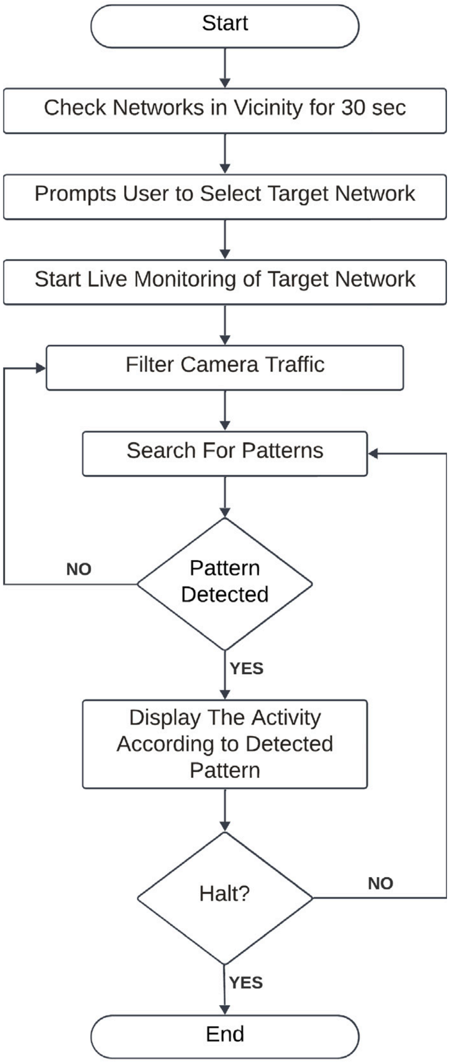

Start
Check Networks in Vicinity for 30 sec
Prompts User to Select Target Network
Start Live Monitoring of Target Network
Filter Camera Traffic
Search For Patterns
Pattern Detected
NO
YES
Display The Activity According to Detected Pattern
Halt?
NO
YES
End

Fig. 4. Workflow of the program for real-time monitoring system.

3.4.4. Capturing target network

Using the BSSID filter, the program uses a targeted approach to gather live traffic from the chosen network. The application effectively filters out irrelevant traffic by utilizing *PyShark's LiveCapture* functionality to ensure that only packets coming from the target network's access point are captured. The crucial parameter for this targeted capture is the access point's unique identity, or BSSID. PyShark is a Python wrapper for the well-known Wireshark network protocol analyzer. It allows for real-time packet capture and analysis, which makes it easier to monitor the target network continuously. With the help of this exact filtering technique, the program is able to focus on the relevant data and capture each packet sent by the target network in real time, producing a reliable and accurate dataset for further analysis stages.

3.4.5. Identifying camera traffic

Using two different filtering approaches, the program isolates traffic from the camera. The first technique assigns manufacturer labels to each network flow by using the OUI, which is the first three bytes of the MAC address. By taking use of the fact that these first bytes are exclusive to the device's maker, this technique makes it possible to precisely identify cameras inside network traffic. The technique can efficiently reduce the amount of data that needs to be further examined by filtering traffic based on the OUI and limiting the packets to those that come from wireless cameras.

The second technique uses the distinct traffic patterns generated by the camera, complementing the initial MAC address-based filtering. In Section 3.2, a detailed examination including the explanation of patterns of the specific sequences defining the traffic of the camera is conducted. Through the examination of these traffic patterns, the algorithm is able to precisely recognize the data from the Xiaomi camera and differentiate it from other network devices. The addition of MAC filtering analysis offers a better level

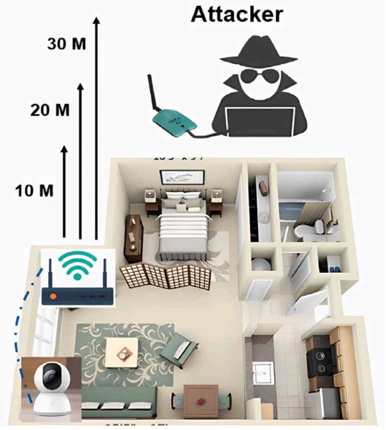


Fig. 5. Experimental attack distance setup.

of precision, guaranteeing that the algorithm separates and concentrates on the relevant camera traffic, even though traffic patterns alone can identify cameras. This combination method improves the real-time monitoring system's accuracy and robustness, making it possible to identify and analyze the Xiaomi camera's network activity with greater efficiency.

3.4.6. Inferring camera activities

Here the study explores the details of network traffic pattern analysis, with a primary focus on recognizing the two types of patterns. The first pattern relates to notifications that the camera generates, which could indicate things like motion or person detection. After identifying the pattern of notifications, the algorithm continues to analyze the information in more depth, examining the packets to ascertain the type of notification. The method distinguishes between alerts for motion detection and those for person detection by examining particular aspects of the pattern. When an event is detected, the system quickly displays the associated activity detection result, giving users instantaneous information about what is happening within the camera's field of vision.

The system also looks for patterns that could indicate live streaming activity that the camera owner has started. The program detects the presence of live streaming activities by watching the traffic flow for indications typical of live streaming sessions, like persistent data transfer rates and repeating packet sequences. User is notified of the ongoing live stream by the program, which instantly shows the relevant information upon detection.

3.4.7. Continue monitoring

Unless the user actively stops it, the program keeps an eye on camera traffic to see whether the camera is performing any further activity. This continuous monitoring makes sure the algorithm is always on the lookout for new activity and occurrences in the network environment. By keeping up this ongoing monitoring, the program makes it easier to identify and analyze camera activity in real time, giving users a thorough grasp of how the device behaves over time. The program's dedication to proactive surveillance is emphasized by its iterative approach, which also makes it possible to respond quickly to any anomalies or emergent occurrences found inside the network.

In the final step, the program terminates its execution upon manual intervention by the user, providing a means to halt the monitoring process and conclude the analysis. This user-initiated termination ensures that the program remains responsive to user needs, offering flexibility and control over the monitoring activities.

4. Results and discussion

This section presents the results of a comprehensive assessment of the system conducted over three distinct experiments. Each experiment involves testing the system with a total of 120 activities, comprising 40 instances each of live stream detection, motion detection, and person detection. These carefully planned tests are intended to evaluate the precision and dependability of the system in recognizing and differentiating between activities of wireless camera. The system is tested at varying distances to evaluate its robustness in real-world conditions and to understand its performance under different signal strength scenarios. The collected data provides a detailed analysis of the system's effectiveness, highlighting its strengths and identifying areas for improvement to enhance its practical applicability.

Table 2
Results of experiment no. 1.

		Predicted			
		Person	Motion	Live stream	Not detected
Actual	Person	36	4	0	0
	Motion	6	34	0	0
	Live Stream	0	0	38	2

4.1. Experimental setup

The experimental setup is maintained in accordance with the previously defined configuration, ensuring consistency and reliability during the testing phase. Building upon the initial setup, three tests are conducted to thoroughly evaluate the functionality and effectiveness of the system. The purpose of each experiment is to evaluate the accuracy and usefulness of the system in various scenarios. The setup was established at three varying distances from the Wi-Fi access point, ranging from close proximity to more extended distances, to evaluate its efficacy in real-world conditions. This intentional variation in distance facilitated the simulation of diverse environmental conditions and deployment scenarios, yielding valuable insights into the system's robustness and adaptability. Fig. 5 presents a visual representation of the experimental setup and conditions.

4.1.1. Experiment no. 1

In the first experiment, the monitoring setup is positioned 10 meters away from the Wi-Fi access point, with a wall obstructing direct line of sight between the two. The objective of this configuration is to mimic a typical inside surveillance situation, where transmission quality and signal intensity can be impacted by obstructions like walls. The signal strength is still rather high, ranging from −20 dB to −28 dB, despite the obstacle. Throughout the trial, the focus was on evaluating the system's performance in identifying various activities in these simulated real-world environments, including person detection, motion detection, and live streaming.

4.1.2. Experiment no. 2

It involves positioning the setup at a greater distance of 40 meters from the Wi-Fi access point, with similar environmental conditions maintained as in Experiment 1. This setup aims to evaluate the system's performance under increased distance from the access point, which could potentially lead to weaker signal strength and increased susceptibility to interference. Despite the greater distance, the signal strength is remained relatively good, ranging between −30 dB to −37 dB throughout the experiment.

4.1.3. Experiment no. 3

This involves testing the system's performance at a distance of 30 meters from the Wi-Fi access point, further challenging the system's ability to detect activities accurately under conditions of increased distance and potentially weaker signal strength. At this distance, the signal strength ranges between −55 dB to −70 dB throughout the experiment, indicating a significant reduction compared to closer proximity scenarios. This setup aimed to evaluate the system's robustness and reliability in real-world scenarios where the Wi-Fi signal may be weaker or subject to interference.

4.2. Results evaluation

In the first experiment, conducted at a distance of 10 meters, the system exhibits good accuracy in recognizing various types of activities. For person detection, the system correctly identifies 36 out of 40 instances. However, in 4 instances, it erroneously classify person detection as motion detection. Similarly, for motion detection, the system accurately classify 34 out of 40 instances, while misclassify 6 instances as person detection. When it comes to live streaming detection, the system shows the highest accuracy, correctly identifying 38 out of 40 instances. There are 2 instances where live streaming are not detected at all. These results highlight the system's robustness in detecting live streaming activities. The results of the first experiment are presented in Table 2, providing a clear overview of the system's performance across the different activity types.

In the second experiment, conducted at a distance of 20 meters, the system's performance exhibits some variations compared to the first experiment. For person detection, the system accurately recognizes 32 out of 40 instances. However, it misclassify 6 instances as motion detection and fails to detect 2 instances. When it came to motion detection, the system maintains a relatively high accuracy, correctly classifying 34 out of 40 instances. It misclassify 5 instances as person detection, and 1 instance of motion detection are undetected. For live streaming detection, the system correctly identifies 36 out of 40 instances, though it fails to detect 4 instances. These results indicate that while the system remains robust, the increased distance does introduce some challenges, particularly in distinguishing between person and motion detection. The detailed outcomes of the second experiment can be observed in Table 3.

In the third experiment, carried out at a distance of 30 meters, the system's accuracy exhibits noticeable declines, highlighting the impact of increased distance on performance. The system accurately recognizes person detection in 28 out of 40 instances. However, 6 instances were misclassified as motion detection, and another 6 instances were not detected at all. For motion detection, the system correctly classify 26 out of 40 instances but misclassify 8 instances as person detection and fails to detect 6 instances. In terms of live streaming detection, the system correctly identify 32 out of 40 instances, with 8 instances going undetected. These results suggest that at 30 meters, the lower signal strength led to significant packet loss, which in turn affects the overall accuracy of the system. The details of the third experiment's results can be found in Table 4.

Table 3
Results of experiment no. 2.

		Predicted			
		Person	Motion	Live stream	Not detected
Actual	Person	32	6	0	2
	Motion	5	34	0	1
	Live Stream	0	0	36	4

Table 4
Results of experiment no. 3.

		Predicted			
		Person	Motion	Live stream	Not detected
Actual	Person	28	6	0	6
	Motion	8	26	0	6
	Live Stream	0	0	32	8

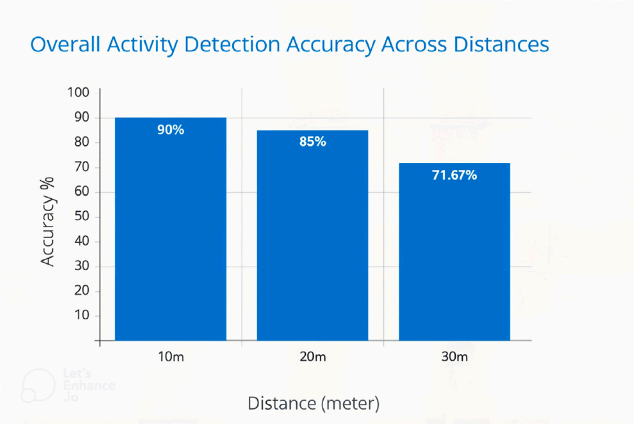


Fig. 6. Overall accuracy graph of the system.

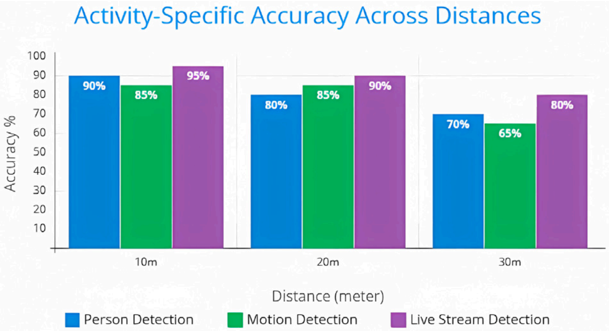


Fig. 7. Activity specific accuracy graph.

4.3. Discussion and analysis

This section discusses the interpretation of the experimental results, focusing on the performance of the system under various conditions. By comparing the outcomes of the three experiments conducted at different distances, the influence of signal strength on the accuracy of activity detection is better understood. Key findings are highlighted, challenges encountered will be addressed, and insights into the factors affecting system performance are provided. Additionally, potential improvements are explored, along with the implications of the results for real-world applications.

4.3.1. Effectiveness of the attack model

The attack model demonstrates varying degrees of effectiveness based on the distance from the target, with overall accuracy rates of 90% at 10 meters, 85% at 20 meters, and approximately 72% at 30 meters. As shown in Fig. 6, the accuracy declines as the distance increases, which is attributed to signal strength degradation and packet loss. These results indicate that the system performs exceptionally well at shorter distances, maintaining high accuracy in detecting activities.

Activity-specific accuracy, depicted in Fig. 7, reveals that person detection accuracy is 90% at 10 meters, 80% at 20 meters, and 70% at 30 meters. Motion detection accuracy is consistent at 85% for 10 and 20 meters but dropped to 65% at 30 meters. For both the 10-meter and 20-meter experiments, this consistency is attributed to the subtle differences between the patterns of motion detection and person detection. In scenarios where packet loss occurs, if the lost packets are those that distinguish between these two activities, the system may misclassify motion detection as person detection and vice versa. In both the 10-meter and 20-meter experiments, a similar amount of crucial packets, which are essential for distinguishing between these activities, are lost. This resulted in the observed uniformity in detection accuracy across these two distances. Live stream detection is the most reliable, with accuracy rates of 95% at 10 meters, 90% at 20 meters, and 80% at 30 meters. These findings suggest that live stream detection is less impacted by distance compared to person and motion detection.

Analysis of the attack model indicates that it is highly effective in short to moderate range scenarios, with a notable decrease in performance at longer distances due to lower signal strength and increased packet loss. This decline highlights the need for strong Wi-Fi signals to maintain high accuracy. Despite these limitations, the model's ability to accurately detect various activities without the need for complex machine learning algorithms demonstrates its practicality and efficiency in real-world applications.

4.3.2. Scalability of the attack model

By utilizing the inherent patterns in Wi-Fi traffic generated by these cameras, the model can be applied to networks with multiple devices, identifying and classifying specific camera activities with high accuracy. The ability to function without relying on sophisticated machine learning algorithms enhances its adaptability. As long as the surveillance notifications are enabled, this approach can scale across diverse settings, from single-family homes to larger buildings, ensuring robust activity inference even amidst numerous other connected devices. This scalability highlights the broader implications of the findings, emphasizing the urgent need for enhanced security measures to safeguard against potential privacy invasions.

4.3.3. Mitigation techniques against attack model

To protect the privacy of users relying on Wi-Fi cameras for home security, one effective strategy, as proposed by [19], is to prevent adversaries from intercepting and collecting smart home network traffic altogether. Implementing methods such as VPN tunneling or other forms of packet header obfuscation can significantly enhance privacy by making it challenging for attackers to identify individual devices within the network. These techniques effectively mask the traffic patterns and metadata that the attack model relies on, thereby thwarting potential attacks and preserving the confidentiality of surveillance activities. By adopting these measures, users can safeguard their smart home environments from unauthorized inference and maintain a higher level of security and privacy.

4.3.4. Comparison with existing techniques

The attack model primarily targets to determine whether someone is present in the house, if not then whether the homeowner is live streaming the camera feed. The study reveals that this information can be inferred without relying on machine learning (ML) models. The distinctiveness of the traffic patterns associated with these activities allows a sophisticated algorithm to detect them effectively. This simplicity is one of the significant achievements of the attack model.

In contrast, existing techniques in the literature review often focus on inferring more granular activities, such as whether a person is sleeping, eating, moving, or engaging in other daily activities. However, an attacker with malicious intent typically needs only to know if someone is in the building and whether the camera feed is being monitored. The model addresses this practical requirement directly, making it highly relevant for real-world security threats. While ML methods offer high accuracy, they do so at the cost of requiring extensive training data and computational power.

This approach demonstrates superior scalability and efficiency, requiring minimal computational resources. It is robust and straightforward for deployment across various environments. Unlike machine learning-based techniques that often necessitate large datasets and frequent retraining to adapt to different settings, this method remains consistently effective with a simple yet powerful algorithm. This advantage enhances the accessibility and feasibility of the attack model for widespread use.

Despite these strengths, both approaches have their limitations. The accuracy of the system depends significantly on signal strength, which could be improved with better hardware and packet restoration techniques. ML methods, while accurate, face challenges with generalization across different environments and the high computational demands for training and inference.

Our proposed non-ML approach prioritizes simplicity and scalability over computational complexity, making it feasible for attack model without extensive resources. Our approach is device-agnostic, requiring only basic traffic pattern analysis to infer activities. However, this simplicity comes with trade-offs. While we achieve high accuracy in controlled environments (up to 90% at 10 m), accuracy declines at longer distances (72% at 30 m) due to packet loss, signal interference, and environmental noise. In more complex network environments with high traffic variability, ML-based models might better adapt by learning from diverse patterns, whereas our approach depends on distinct, repeatable packet structures. Additionally, ML methods could infer more granular activities, whereas our technique is limited to detecting notifications and broad activity types. Despite these trade-offs, the advantage of our method lies in its real-time usability, computational efficiency, and ease of deployment, making it a viable attack model in many real-world scenarios. Future work could explore hybrid approaches that balance simplicity with adaptive accuracy, such as lightweight ML models that enhance packet pattern recognition without excessive computational overhead.

5. Conclusion

This article presents an innovative and efficient method for extracting crucial information from the encrypted Wi-Fi traffic of wireless cameras, enabling accurate identification of activities such as person detection, motion detection, and live streaming without relying on complex machine learning (ML) models. Unlike ML-based approaches like Liu et al. [18], which achieve 85%–95% accuracy but require extensive datasets and computational resources, our method attains 90% accuracy at 10 meters by exploiting deterministic traffic patterns (e.g., 30–33 packet bursts during notifications). Similarly, while statistical techniques such as Apthorpe et al. [19] and Huang et al. [20] report 70%–85% accuracy for activity inference, their scalability is limited by encryption variability and device diversity. Our approach eliminates these dependencies, offering comparable accuracy with minimal computational overhead.

The model's simplicity and scalability make it adaptable to diverse environments, from single-family homes to large buildings, outperforming device-specific profiling methods [21,24] in real-time practicality. However, like traffic-based techniques [19,22], its accuracy declines at longer distances (e.g., 72% at 30 m) due to packet loss, underscoring the impact of signal attenuation. These findings emphasize the urgent need for enhanced security measures in Wi-Fi camera systems, as even encrypted traffic leaks metadata exploitable by lightweight, non-ML methods.

By prioritizing real-world applicability over theoretical granularity, this work bridges a critical gap between resource-intensive ML models and oversimplified statistical analyses. Future efforts could integrate signal-strength-aware algorithms or randomized packetization to mitigate metadata leakage. Ultimately, this research underscores the dual imperative for manufacturers to redesign IoT encryption protocols and for policymakers to address the privacy risks inherent in modern surveillance technologies.

Future research will focus on addressing these limitations through advanced packet restoration techniques and integrating lightweight machine learning models to improve generalization across diverse environments.

CRediT authorship contribution statement

Muhammad Bilal Rasool: Writing – original draft, Methodology, Data curation, Conceptualization. **Uzair Muzamil Shah:** Visualization, Software, Methodology, Formal analysis, Conceptualization. **Mohammad Imran:** Validation, Supervision, Software, Resources, Project administration, Investigation, Conceptualization. **Daud Mustafa Minhas:** Writing – review & editing, Visualization, Investigation, Formal analysis. **Georg Frey:** Writing – review & editing, Validation, Investigation, Funding acquisition, Formal analysis.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: This research received no external funding, and the APC was funded by Universität des Saarlandes, Germany.

Data availability

No data was used for the research described in the article.

References

[1] S.M. Beyer, B.E. Mullins, S.R. Graham, J.M. Bindewald, Pattern-of-life modeling in smart homes, IEEE Internet Things J. (2018) 1–8, http://dx.doi.org/10.1109/JIOT.2018.2840451.

[2] A. Sivanathan, H.H. Gharakheili, F. Loi, A. Radford, C. Wijenayake, A. Vishwanath, V. Sivaraman, Classifying iot devices in smart environments using network traffic characteristics, IEEE Trans. Mob. Comput. (2018) 1, http://dx.doi.org/10.1109/TMC.2018.2866249.

[3] Market analysis report, 2021, Available online: https://www.grandviewresearch.com/industry-analysis/smart-home-security-camera-market. (Accessed 15 May 2021).

[4] Yushi Cheng, Xiaoyu Ji, Tianyang Lu, Wenyuan Xu, On detecting hidden wireless cameras: A trafc patternbased approach, IEEE Trans. Mob. Comput. 19 (2019) (2019) 907–921, 4.

[5] T.S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, et al., Millimeter wave mobile communications for 5G cellular: It will work! IEEE Access 1 (2013) 335–349.

[6] J.G. Andrews, S. Buzzi, W. Choi, S.V. Hanly, A. Lozano, A.C.K. Soong, J.C. Zhang, What will 5G be? IEEE J. Sel. Areas Commun. 32 (6) (2014) 1065–1082.

[7] S. Yadav, A. Yadav, A. Shrestha, Network security and privacy challenges in internet of things: A comprehensive study, in: 2021 International Conference on Sustainable Technologies for Industry 4.0, STI, IEEE, 2021, pp. 1–6.

[8] Amazon Echo. https://www.amazon.com/Amazon-Echo-Bluetooth-Speaker-with-WiFi-Alexa/dp/B00X4WHP5E.

[9] Y. Li, S. Zhu, S. Zeadally, Wi-Fi attacks and defense mechanisms: A comprehensive survey, IEEE Commun. Surv. Tutor. 23 (3) (2021) 2010–2044.

[10] S. Zhong, K. Xu, M. Jia, Y. Zhang, Vulnerability analysis and countermeasures of home Wi-Fi camera, in: 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), IEEE, 2019, pp. 1–7.

[11] A. El-Mougy, M. Ismail, H. Nassar, Wireless intrusion detection and prevention system: A comprehensive review, IEEE Commun. Surv. Tutorials 23 (3) (2021) 2467–2508.

[12] Tadayoshi Kohno, Andre Broido, Kimberly C. Claffy, Remote physical device fingerprinting, IEEE Trans. Dependable Secur. Comput. 2 (2) (2005) 93–108.

[13] A. Goldsmith, S.A. Jafar, N. Jindal, S. Vishwanath, Capacity limits of MIMO channels, IEEE J. Sel. Areas Commun. 21 (5) (2009) 684–702.

[14] C. Chen, Y. Gong, K. Zhang, Y. Zhang, A survey of network security in wireless networks, J. Netw. Comput. Appl. 96 (2018) 1–20.

[15] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Comput. Netw. 54 (15) (2010) 2787–2805.

[16] M. Centenaro, L. Vangelista, A. Zanella, M. Zorzi, Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios, IEEE Wirel. Commun. 23 (5) (2016) 60–67.

[17] P.P. Ray, Security for internet of things: A survey of existing protocols and open research issues, IEEE Commun. Surv. Tutor. 20 (3) (2018) 2223–2285.

[18] Xiaolong Liu, et al., 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems, HPCC/SmartCity/DSS, in: Inferring behaviors via encrypted video surveillance traffic by machine learning, IEEE, 2019.

[19] Noah Apthorpe, et al., Spying on the smart home: Privacy attacks and defenses on encrypted iot traffic, 2017, arXiv preprint arXiv:1708.05044.

[20] Qianyi Huang, et al., Rethinking privacy risks from wireless surveillance camera, ACM Trans. Sens. Netw. 19 (3) (2023) 1–21.

[21] M. Alyami, I. Alharbi, C. Zou, Y. Solihin, K. Ackerman, Wi-Fi-based IoT devices profiling attack based on eavesdropping of encrypted wi-Fi traffic, in: 2022 IEEE 19th Annual Consumer Communications & Networking Conference, CCNC, Las Vegas, NV, USA, 2022, pp. 385–392, http://dx.doi.org/10.1109/CCNC49033.2022.9700674.

[22] A.J. Pinheiro, J.D.M. Bezerra, C.A. Burgardt, D.R. Campelo, Identifying IoT devices and events based on packet length from encrypted traffic, Comput. Commun. 144 (2019) 8–17.

[23] Abbas Acar, et al., Peek-a-boo: I see your smart home activities, even encrypted! in: Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks, 2020.

[24] Shuaike Dong, et al., Your smart home can't keep a secret: Towards automated fingerprinting of iot traffic, in: Proceedings of the 15th ACM Asia Conference on Computer and Communications Security, 2020.

[25] Jide S. Edu, Jose M. Such, Guillermo Suarez-Tangil, Smart home personal assistants: a security and privacy review, ACM Comput. Surv. 53 (6) (2020) 1–36.

[26] Daniele Mari, et al., Looking through walls: Inferring scenes from video-surveillance encrypted traffic, in: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2021.

[27] Jakub Markiewicz, et al., The example of using the xiaomi cameras in inventory of monumental objets-first results, Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci. 42 (2017).

[28] He-Jun Lu, Yang Yu, Research on WiFi penetration testing with Kali Linux, Complexity 2021 (2021) 1–8.

[29] Andy Sun, et al., A castle of glass: Leaky iot appliances in modern smart homes, IEEE Wirel. Commun. 25 (6) (2018) 32–37.

[30] A. Masiukiewicz, V. Tarykin, V. Podvornyi, Security threats in Wi-Fi networks, Eng. Sci. 1 (3) (2016) 6–11.

[31] Chris Sanders, Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems, No Starch Press, 2017.

[32] Aayush Majumdar, Shruti Raj, T. Subbulakshmi, ARP poisoning detection and prevention using Scapy, J. Phys.: Conf. Ser. 1911 (1) (2021) IOP Publishing.

[33] Adrian Dsouza, et al., Real time network intrusion detection using machine learning technique, in: 2022 IEEE Pune Section International Conference, PuneCon, IEEE, 2022.