IoT

MDPI

Article

Weaponized IoT: A Comprehensive Comparative Forensic Analysis of Hacker Raspberry Pi and PC Kali Linux Machine

Mohamed Chahine Ghanem [1,2,*], Eduardo Almeida Palmieri [1], Wiktor Sowinski-Mydlarz [1], Sahar Al-Sudani [1] and Dipo Dunsin[1]

[1] Cyber Security Research Centre, London Metropolitan University, London, N78DB, UK; eda0203@my.londonmet.ac.uk (E.A.P.); w.sowinskimydlarz@londonmet.ac.uk (W.S.-M.); d.dunsin@londonmet.ac.uk (D.D.); s.alsudani@londonmet.ac.ukk (S.A.-S.)
[2] Institute for Cybersecurity, University of Liverpool, Liverpool, L69 3BX, UK
* Correspondence: mohamed.chahine.ghanem@liverpool.ac.uk

Abstract: The proliferation of Internet of Things (IoT) devices presents significant challenges for cybersecurity and digital forensics, particularly as these devices have become increasingly weaponised for malicious activities. This research focuses on the forensic analysis capabilities of Raspberry Pi devices configured with Kali Linux, comparing their forensic capabilities to conventional PC-based forensic investigations. The study identifies key gaps in existing IoT forensic methodologies, including limited tool compatibility, constrained data retention, and difficulties in live memory analysis due to architectural differences. The research employs a testbed-based approach to simulate cyberattacks on both platforms, capturing and analysing forensic artefacts such as system logs, memory dumps, and network traffic. The research findings reveal that while traditional PCs offer extensive forensic capabilities due to superior storage, tool support, and system logging, Raspberry Pi devices present significant forensic challenges, primarily due to their ARM architecture and limited forensic readiness. The study emphasises the need for specialised forensic tools tailored to IoT environments and suggests best practices to enhance forensic investigation capabilities in weaponised IoT scenarios. This research contributes to the field by bridging the gap between theoretical frameworks and real-world forensic investigations, offering insights into the evolving landscape of IoT forensics and its implications for digital evidence collection, analysis, and forensic readiness.

Keywords: IoT digital forensics; Raspberry Pi; single-board computers; cybercrime; digital forensics; Linux; Kali; hacking; digital investigation; weaponized IoT; Exterro FTK; volatility; magnet AXIOM

Academic Editor: Amiya Nayak

Received: 3 January 2025
Revised: 28 February 2025
Accepted: 6 March 2025
Published: 7 March 2025

Citation: Ghanem, M.C.; Almeida Palmieri, E.; Sowinski-Mydlarz, W.; Al-Sudani, S.; Dunsin, D. Weaponized IoT: A Comprehensive Comparative Forensic Analysis of Hacker Raspberry Pi and PC Kali Linux Machine. *IoT* **2025**, *6*, 18. https://doi.org/10.3390/iot6010018

1. Introduction

IoT devices continue to proliferate at an unprecedented rate, and it is forecast that by 2025, they will make up more than two-thirds of an estimated 41.6 billion internet-connected devices [1]. These devices, ranging from smart home gadgets to complex industrial systems, form a core part of modern ecosystems but also pose new challenges for cybersecurity and forensic investigations. Compact, low-cost single-board computers, such as the Raspberry Pi, exemplify this duality. While these devices are in general deployed as central components in IoT environments for the collection, control, and analysis of sensor data, praised for their high processing power, low price, and user-friendliness, they can easily be utilized by cybercriminals when loaded with tools like Kali Linux as

machines for hacking. This dual-purpose nature underlines the requirement for specialist IoT forensic techniques.

IoT forensics is a relevant and relatively new area; adding new dimensions in cybercrime presents challenges for digital forensics. IoT forensics refers to the identification, collection, and preservation and analysis of digital evidence from IoT devices. Its importance lies in providing insights into how devices communicate with each other, their network behaviour, and user activities—the key elements that can be used to solve incidents such as data breaches, cyberattacks, or even physical crimes. IoT ecosystems generate vast volumes of data, and as the systems become ubiquitous, it will only increase the necessity for having robust forensic methodologies and techniques in place [2].

IoT forensics includes several important subdomains: device, live, network, and cloud forensics. IoT devices—often, though not always, called "things"—pose challenges to forensics. Some have permanent storage with recognizable file systems and formats, so traditional forensics works. Yet, others are based on proprietary file systems or do not have permanent memory at all, which complicates evidence collection. Power supply limitations, low RAM, and real-time data transfer worsen the situation for live forensics. The data transferred over the networks can be encrypted. Moreover, IoT data, when processed in cloud environments, often resides at remote and sometimes unknown geographical locations, making retrieval and analysis even more complex [3].

Forensic investigations of IoT devices are further complicated by the volatility of IoT data, the intricate architectures of IoT ecosystems, and the integration of these devices into larger network infrastructures. There are also legal and ethical issues to contend with, including privacy and ensuring a proper chain of custody.

Figure 1 shows the sub-components of IoT forensics and corresponding sources of artefacts, which include the following:

- Criminal-operated Linux systems (e.g., command-and-control servers).
- Abused or misused Linux systems (e.g., by suspect users).
- Imaged systems (e.g., dead disks).
- Standalone artefacts from Linux distributions.
- Raspberry Pi devices running Kali Linux.
- Metapackages from other platforms.

The uniqueness of IoT forensic investigations demands specialized techniques and tools tailored to the constraints and complexities of these environments. The diversities of the evidence sources are depicted in Figure 1, showing diversity and complexity; these sources need to be handled with care to analyse the artefacts for accurate and reliable output [4].

Lastly, even though the defence-in-depth approach is obviously necessary in securing an IoT ecosystem, it opens up new complexities for forensic investigations. This involves the application of different mechanisms to protect against possible attacks. However, these defence layers are often compromised due to human errors, misconfiguration, and built-in system vulnerabilities. Since cyber threats have been diversifying and intensifying, a growing number of security controls and measures compound the challenges faced by forensic analysts [5]. These require not only technical expertise but also a deep understanding of the peculiarities of IoT environments.

This research investigates the forensic processes involved in analysing a Raspberry Pi configuration using Kali Linux as a tool for IoT-based cyberattacks. The study seeks to highlight the differences between forensic investigations conducted on conventional computers and those on IoT devices, particularly in terms of artefact extraction, data retention, and forensic tool compatibility. Additionally, it examines the challenges faced in IoT forensics and proposes best practices to enhance forensic readiness for such attacks.

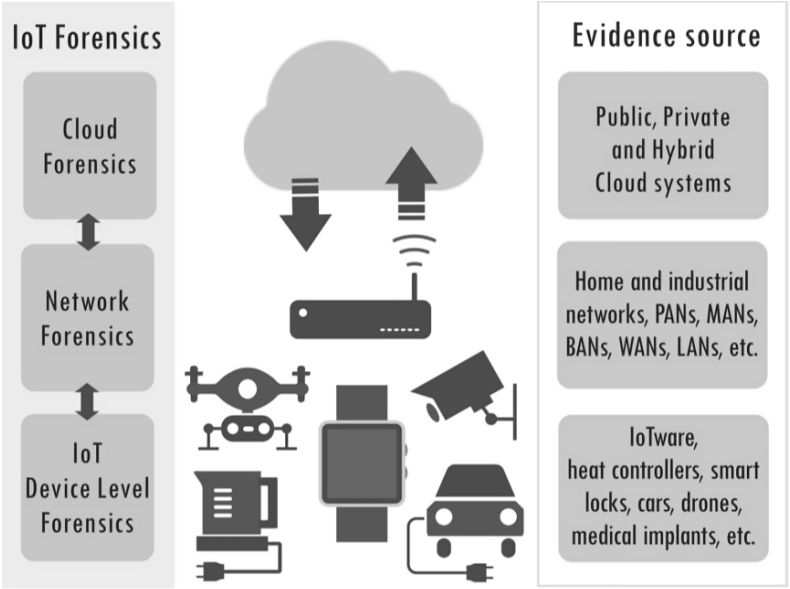

Figure 1. IoT forensics domain and artefact categories [4].

The rapid proliferation of IoT devices has introduced significant forensic challenges, necessitating advanced investigative approaches. IoT forensics presents complex challenges across several dimensions, starting with evidence source identification, as evidence is often distributed across a variety of interconnected devices with different protocols and data storage formats. This dispersion complicates forensic investigations, making it crucial to develop efficient methods for locating and correlating evidence. Moreover, evidence acquisition or forensic imaging is inherently difficult, given IoT devices' limited storage, proprietary data formats, and reliance on cloud-based storage, which complicates access [6]. The compromise of forensic integrity without robust acquisition methods underscores the need for tailored forensic frameworks. Additionally, traditional forensic tools and techniques are insufficient for IoT-specific needs, as they are not designed to handle the unique architectures and data formats of IoT devices, necessitating the use of specialised tools. The absence of standardised forensic methodologies for the IoT further exacerbates investigative difficulties, highlighting an urgent research gap in developing adaptive forensic solutions. These factors collectively underscore the importance of customised forensic strategies and international legal collaboration to effectively address the complexities of IoT investigations, motivating research efforts to establish standardised frameworks and cross-jurisdictional cooperation in IoT forensics.

Our research questions are formulated as follows:

- **RQ1**: How do forensic processes differ between conventional computers and IoT devices such as Raspberry Pi devices?
- **RQ2**: What are the key differences in terms of meaningful forensic artefacts between conventional computers and IoT devices?
- **RQ3**: What are the current challenges and limitations in IoT forensics and possible best practices to implement to overcome these challenges?

The reminder of this paper is organised as follows: Section 2 presents a literature review on key aspects of IoT forensics. It explores the key challenges and opportunities within the IoT environment, particularly focusing on cybersecurity and deterrence strategies. It critically reviews existing research on ethical hacking and penetration testing and provides a brief comparison with related works. Section 3 describes the methodology employed in this research, focusing on the creation of a test-bed that simulates environments where conventional PCs and the Raspberry Pi operate together, as well as providing a

detailed account of the comparative forensics investigation performed. Section 4 presents the research findings and discusses the differences in forensic capabilities between conventional PCs and the Raspberry Pi. Section 5 concludes the paper by emphasizing the significant differences in forensic capabilities between traditional PCs and Raspberry Pi devices, particularly highlighting limitations of the Raspberry Pi such as storage and compatibility issues. It recommends conventional PCs as a more robust and reliable platform for forensic applications.

2. Literature Review

As IoT devices proliferate across various domains, understanding the unique characteristics of IoT forensics has become imperative for effective investigation and incident response. This section highlights the scope of IoT forensics. IoT forensics encompasses a broad range of activities and challenges related to investigating and analysing digital evidence from Internet of Things (IoT) devices and systems. The scope of IoT forensics extends beyond traditional digital forensics practices to address the unique challenges posed by interconnected, heterogeneous, and pervasive IoT environments [7]. The Internet of Things (IoT) is gaining popularity, and numerous sectors have drawn attention to the topic of IoT security and forensics. Research efforts on IoT security and forensics are extensive and cover a wide range of topics, from the theoretical underpinnings of cybersecurity to the practical challenges of securing and investigating IoT devices [8]. The use of low-cost, portable tools like the Raspberry Pi has emerged as a recurring theme, offering both opportunities and challenges in the field of IoT forensics. Future research should continue to explore these areas, particularly focusing on the development of more robust forensic tools and methodologies tailored to the unique challenges posed by IoT environments. This literature review integrates key studies that explore various aspects of IoT security, digital forensics, and the use of low-cost portable tools like the Raspberry Pi for these purposes. It also offers a thorough examination of existing research on the use of Raspberry Pi in hacking and forensic analysis. It examine the performance, capabilities, and vulnerabilities of these devices within IoT environments, highlighting the crucial role of IoT forensics in criminal investigations and the importance of strong security measures [9]. The uniqueness of IoT forensics can be summarized in the following points:

1. **Device diversity:** IoT devices come in various forms, including single-board computers (SBCs), sensors, actuators, wearables, smart home appliances, and industrial controllers making the task of addressing the diversity of device types, architectures, communication protocols, and operating systems challenging.
2. **Data acquisition:** Retrieving data from IoT devices while preserving their integrity and ensuring admissibility in legal proceedings is complex with many challenges, such as accessing data stored in volatile memory, retrieving logs and configuration settings, and capturing network traffic.
3. **Distributed nature:** IoT environments involve numerous geographically distributed devices, making data collection and analysis challenging, especially with real-time data generation.
4. **Scalability issues:** The vast number of devices and data in IoT systems demands new forensic approaches to efficiently process and analyse large-scale information.
5. **Heterogeneous protocols:** IoT devices use various communication protocols, requiring forensic experts to understand and analyse diverse and often complex interactions.
6. **Privacy and legal concerns:** IoT devices collect sensitive data, raising privacy issues. Forensic investigations must navigate legal frameworks to ensure evidence is admissible without violating privacy rights.

Analysing data obtained from IoT devices involves examining logs, event traces, metadata, communication patterns, and potentially large volumes of sensor data. Therefore, investigators need tools and techniques to process, interpret, and correlate diverse data sources for reconstructing events and identifying evidence. Network forensics refers to IoT devices' communication over wireless and wired networks; this presents challenges for capturing, analysing, and reconstructing network traffic [1]. Therefore, investigators must consider encryption, encapsulation, and fragmentation mechanisms used in IoT communication protocols. Many IoT deployments leverage cloud and edge computing platforms for data processing, storage, and analytics. Forensic investigations may involve accessing data stored on remote servers, analysing data streams at the edge, and tracing data flows across distributed architectures.

A study conducted in 2018 in ref. [1] explored the topic of cybersecurity and deterrence within IoT environments, focusing on strategies to prevent cyberattacks, particularly those orchestrated by nation states. In it, the authors highlight the key differences between cyber weapons and conventional military tools, delving into the motivations that drive cyber operations. Additionally, the authors assess the effectiveness of various deterrence strategies. While their research does not specifically address IoT forensics, it lays the foundation for understanding the broader cybersecurity landscape, which is vital for contextualizing the unique challenges faced in IoT environments. Ref. [10] provides a comprehensive survey of the challenges and methodologies in IoT forensics, emphasizing critical areas such as the establishment of data inclusion and exclusion criteria, the automation of forensic processes, and the integration of forensic capabilities into device design, referred to as "forensics by design". The research also addresses the usability of forensic tools, the complexities involved in shutting down IoT devices for analysis, the implications of service-level agreements (SLAs) on data access, and the privacy risks associated with encryption and anti-forensic techniques. This study is essential for understanding the intricate landscape of IoT forensics and the various obstacles that practitioners encounter in their investigations.

In a separate investigation, ref. [11] explored the field of ethical hacking and penetration testing, highlighting the advantages of using low-cost, portable hardware such as the Raspberry Pi. The authors provide a thorough overview of ethical hacking, covering essential definitions, techniques, and the practical application of various tools, particularly using a Raspberry Pi for tasks like reconnaissance and remote penetration testing. Integrating theoretical insights and hands-on practices, this study is a valuable resource for understanding how portable devices can enhance cybersecurity efforts and forensic investigations. A study conducted in ref. [12] investigated the vulnerabilities using a Raspberry Pi 4 running Raspberry Pi OS to simulate attacks using Kali Linux and various automated tools. Their research revealed significant security concerns inherent to IoT devices, emphasising the critical need for robust security measures to prevent potential exploitation. The methodology outlined in their work provided a comprehensive discussion of the practical challenges faced in securing IoT environments, highlighting the complexities involved in protecting these devices from cyber threats.

Ref. [8] investigated the creation of a low-cost, portable digital forensic imaging tool using a Raspberry Pi. The primary objective of the research was to develop an affordable imaging solution capable of effectively collecting and analysing digital evidence. This work is especially significant to IoT forensics, where the need for cost-effective tools is critical due to the extensive variety and prevalence of IoT devices in various environments. In [13], the researchers conducted an evaluation and comparison of two open-source intrusion detection systems (IDSs) operating on a Raspberry Pi 2 (Model B). The primary objective of their research was to assess the suitability of these systems for deployment in cost-sensitive network environments. This investigation holds significant importance for IoT forensics,

emphasising the need for effective intrusion detection mechanisms in resource-constrained scenarios, where affordable hardware plays a crucial role.

The research conducted in [14] examined the security vulnerabilities found in two commercial drones, utilising the Raspberry Pi as an automated tool for their analysis. This study not only highlights the significant security weaknesses inherent in these drones but also demonstrates the potential for exploiting such vulnerabilities. By shedding light on these risks, the research contributes valuable insights to the broader field of IoT forensics by emphasizing the dangers associated with the increasing integration of IoT devices across various sectors. A summary of related work, aspects covered, and techniques used is presented in Table 1.

The literature identifies a notable gap in the field regarding comparative studies of forensic processes between Raspberry Pi and traditional PCs. While many studies focus on specific aspects of IoT forensics, there is a lack of research that directly compares the forensic capabilities and challenges of these two platforms. The literature also highlights ongoing debates regarding the effectiveness of various forensic tools and methodologies when applied to IoT devices [15].

Table 1. Comparative Summary of related works.

Reference	Year	IoT	Digital Forensics	Offensive Security	Technique and Approach
[10]	2020	✓	✓	✓	Surveyed challenges, approaches, and open issues in the field of IoT forensics, research broadly highlighted differences and similarities between mobile and IoT forensics, and tackled forensics by design and digital forensics as a service (DFaaS).
[11]	2017	✓	✗	✓	Comprehensive overview of ethical hacking practices, emphasizing the use of low-cost, portable hardware like the Raspberry Pi. Defines ethical hacking, penetration testing, reconnaissance techniques, and remote penetration testing with the RPI, combining theoretical and practical aspects.
[12]	2022	✓	✓	✗	The article focuses on demonstrating the vulnerability of IoT devices using a Raspberry Pi 4 with Raspberry Pi OS. Attacks with Kali Linux and automated tools are employed, highlighting the security concerns associated with IoT devices. The methodology of executing the attacks is discussed, emphasising the importance of securing IoT devices to prevent exploitation.
[8]	2021	✓	✗	✓	The paper focuses on developing a low-cost, and portable digital forensic imaging tool using the RPI. The goal is to create an image that can be used and analysed as reliable evidence.
[13]	2015	✓	✓	✗	Focus on evaluating and comparing the performance, efficiency, and efficacy of two open-source intrusion detection systems (IDSs) running on the Raspberry Pi 2 (Model B). Aim to determine their suitability for use in cost-sensitive network environments.
[14]	2019	✓	✓	✗	Identify and exploit vulnerabilities in two commercial drones. Aim to demonstrate the security weakness present in these drones by using the Raspberry Pi as an automated tool to interact with the drones.

Table 1. *Cont.*

Reference	Year	IoT	Digital Forensics	Offensive Security	Technique and Approach
[7]	2024	✓	✓	✗	A comprehensive review of IoT forensic techniques, categorising them into device, network, and cloud forensics. This comprehensive review identifies the challenges and gaps present in current methodologies and explores the necessity for customised forensic tools specifically designed for IoT environments.
[16]	2023	✓	✓	✗	Proposes an improved IoT forensic model to address the interconnectivity of IoT devices. The model primarily concentrates on enhancing forensic readiness and addressing the challenges associated with distributed IoT ecosystems.
[17]	2019	✓	✓	✗	Provides a taxonomy of IoT forensic techniques and highlights open challenges. Discusses the importance of addressing heterogeneity and scalability in IoT forensic investigations.
[18]	2018	✓	✓	✗	Introduces privacy-aware forensic methodologies for IoT devices. Focuses on cooperative digital investigations and the balance between privacy and forensic needs.
[19]	2019	✓	✓	✗	Reviews challenges in IoT forensics and explores the role of artificial intelligence in addressing these challenges. It offers a forward-looking perspective on IoT forensic methodologies.

2.1. Comparison with Existing Research

The increasing weaponisation of IoT devices presents significant challenges for digital forensics, necessitating targeted research on forensic methodologies tailored for these emerging threats. While existing studies explore broad conceptual frameworks for IoT forensics, this study contributes by providing a targeted forensic analysis of an easily configurable attack system using a Raspberry Pi with Kali Linux. Previous research has emphasised the complexities of IoT forensics due to the diverse architectures, communication protocols, and storage limitations of these devices [10]. However, there remains a gap in practical forensic methodologies that address specific attack scenarios involving low-cost, readily available hardware like the Raspberry Pi.

This study bridges that gap by examining the forensic implications of Raspberry Pi-based IoT attacks and comparing forensic evidence collected from both a Raspberry Pi and a traditional PC. Unlike prior studies that primarily focus on ethical hacking and penetration testing using the Raspberry Pi [11], this research takes a forensic approach by analysing digital artefacts left behind after simulated attacks. Furthermore, while research by [12] explored security vulnerabilities in IoT devices using Raspberry Pi and Kali Linux, their study focused on attack execution rather than forensic investigation. In contrast, this research contributes directly to the field of IoT forensics by evaluating the evidential value of Raspberry Pi artefacts and identifying forensic challenges in data retrieval, log analysis, and network forensics.

Moreover, the existing literature has discussed the forensic challenges posed by IoT environments, such as limited storage, a lack of standardisation, and the volatility of IoT data [3]. Ahmed et al. [7] provide a comprehensive review of IoT forensic techniques, challenges, and future directions, emphasising the need for tailored forensic tools in IoT environments. Their work highlights gaps in existing methodologies, reinforcing the importance of studies like this one in addressing real-world forensic challenges. In the

same way, ref. [16] suggests a better forensic model that takes into account how IoT devices can connect to each other. This is an important factor to consider when looking into IoT-based attacks and finding evidence on multiple connected devices.

While studies have highlighted the importance of forensic-by-design principles [1], there remains a lack of empirical research demonstrating practical forensic techniques for IoT devices configured for malicious purposes. This study looks at the differences between Raspberry Pi devices and traditional PCs in terms of forensic evidence collection. It highlights important differences and suggests best practices for tackling challenges in analysing IoT devices. Additionally, while [15] acknowledged the role of forensics in IoT applications, their study focused on a generalised investigation model without a detailed analysis of specific IoT hardware. Similarly, the research in [20] discussed forensic methodologies for the Raspberry Pi but did not compare the forensic artefacts with traditional computing devices.

This study builds upon their findings by providing a structured forensic approach tailored to IoT-based attacks and reinforcing the necessity for specialised forensic tools and methodologies. Yaqoob et al. [17] offer a foundational perspective on IoT forensics advancements, a taxonomy, and open challenges, which directly inform the forensic strategies used in this research. Furthermore, ref. [18] discusses privacy-aware forensic methodologies, a critical consideration when handling IoT-related evidence to ensure compliance with ethical and legal standards. Alenezi et al. [19], further highlight the challenges in IoT forensics and explore the role of artificial intelligence in addressing them, providing a forward-looking perspective on the field.

As a result of highlighting the forensic implications of weaponised IoT devices, this study enhances the existing body of knowledge and serves as a practical guide for forensic investigators. It identifies critical gaps in evidence collection, highlights limitations in existing forensic tools, and proposes recommendations for improving IoT forensics, including the need for live memory analysis techniques and specialised forensic frameworks for IoT environments.

2.2. Table 1: Summary of Related Works

Table 1 presents a structured summary of existing research on IoT forensics, cybersecurity, and offensive security techniques, effectively outlining key studies and their methodologies. This table shows the forensic methods used in each study and points out their main limitations and gaps. This, in turn, aids in elucidating the similarities and differences between this research and previous studies. Organising the studies according to their main topics—like IoT security weaknesses, forensic methods, and offensive security—makes them easier to read and allows for better comparisons. An additional column directly contrasting the scope of previous studies with the methodology applied in this research further emphasizes the study's unique contributions. These refinements enhance the clarity and depth of the comparative analysis, making it more informative for forensic investigators and researchers.

3. Methodology

The rise of Internet of Things (IoT) devices has introduced significant challenges to traditional forensic investigations, which are often tailored to conventional computing environments. This study addresses these challenges by creating a testbed that simulates environments where conventional PCs and IoT devices, like the Raspberry Pi, coexist. By simulating various cyberattacks and analysing data such as network traffic, memory dumps, and system logs, the research compares forensic processes between these plat-

forms. The findings highlight the strengths and limitations of existing forensic tools in IoT environments, emphasizing the need for specialized approaches for IoT forensics.

3.1. Testbed Design

The forensic investigation in this study required a meticulously designed testbed to evaluate and compare forensic processes between conventional computers and IoT devices. The testbed included both a traditional PC and a Raspberry Pi, configured to simulate a real-world environment where these devices coexist. The PC used was equipped with Intel chip, 8 GB RAM and a 120 GB SSD, running Kali Linux 2023.3. The Raspberry Pi 5, representing an IoT device, featured 5 GB RAM and a 32 GB SD card, also running Kali Linux 2023.3. Both devices were connected to a common Wi-Fi network using a Nokia HA-140W-B router. To generate relevant forensic data, four distinct cyberattack scenarios were simulated on both the PC and the Raspberry Pi. These attacks included Windows 7—EternalBlue, PowerShell-Empire, Windows 10—Multi/Handler with Msfvenom payload, and Koadic Framework. Network traffic during these attacks was captured using Wireshark, ensuring comprehensive data for forensic analysis. Figure 2 depicts the key components and processes of the testbed design and implementation.

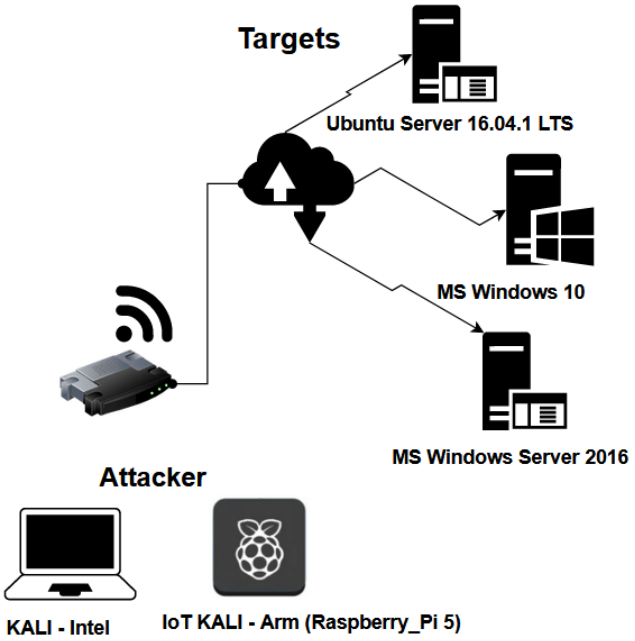

Figure 2. Testbed design and implementation.

3.2. Experimental Setup and Comparative Analysis

We compared traditional PCs and Raspberry Pi devices to show the differences in their forensic abilities, such as which tools they can use, how they store data, memory testing, and log retrieval. This evaluation offers clear insights into the challenges and benefits of forensic investigations in various computing settings, promoting a more careful and data-based method, as illustrated in Algorithm 1.

Algorithm 1 Systematic Forensic Analysis on Linux OS Using FTK and UFED

1: **Initialisation**: Pre-Forensic Preparation Tasks (write-blocking, acquisition and analysis tools (FTK, UFED)).
2: **Step 1: Evidence Acquisition**
   1. Create a forensic image of the target Linux OS:
      - Use FTK Imager to create disk images (E01, RAW/DD).
      - Hash the image using MD5/SHA-256 for integrity.
   2. Acquire volatile memory (if applicable):
      - Use LiME (Linux Memory Extractor) for RAM capture.
      - Validate memory dump integrity using hashing.
3: **Step 2: File System Analysis**
   1. Load the forensic image in FTK.
   2. Identify file system type (Ext4, XFS, Btrfs).
   3. Recover deleted files using FTK's file signature analysis.
   4. Extract system logs from `/var/log`:
      - Analyze `auth.log`, `syslog`, and `bash_history`.
4: **Step 3: Network and Communication Analysis**
   1. Extract network logs using Wireshark and FTK:
      - Capture packet data (`.pcap` files).
      - Identify unauthorized SSH, VPN, or FTP connections.
   2. Analyze user credentials:
      - Extract `/etc/passwd`, `/etc/shadow` for user details.
5: **Step 4: Mobile & IoT Device Data Extraction**
   1. Use UFED to analyse connected Linux-based mobile/IoT devices.
   2. Extract application logs (WhatsApp, Telegram, emails).
   3. Analyze timestamps and metadata.
6: **Step 5: Reporting and Case Documentation**
   1. Generate a forensic report in FTK:
      - Include extracted logs, timestamps, file hashes.
      - Validate findings by cross-verifying with original data.
   2. Export the report in a legally admissible format (PDF, CSV).

3.3. Dataset Elaboration

Various types of data were collected from both devices during the simulations to ensure a thorough forensic investigation. Network traffic data, including all incoming and outgoing packets, were captured and saved in pcap format for later analysis. Memory dumps were obtained using LiME (Linux Memory Extractor) for the Raspberry Pi and Microsoft AVML for the PC. These memory dumps were crucial for analysing the processes and system states during the attacks.

Additionally, system logs, such as event, application, and security logs, were collected from both devices to provide context to the network traffic and memory data. File system snapshots were also taken before and after the attack simulations, allowing for the identification of any changes made by the attackers.

3.4. Data Capture

The forensic data capture process involved both forensic imaging and live RAM dumps. FTK Imager was used to create forensic images of the PC's hard drive and the Raspberry Pi's SD card, providing a complete snapshot of the data at the time of acquisition. For live data capture, RAM dumps were obtained using LiME for the Raspberry Pi and Microsoft AVML for the PC. A forensic analysis of RAM dumps was conducted

using tools such as LiME (Linux Memory Extractor) and AVML (Azure Volatile Memory Library) to extract system processes, active network connections, and in-memory artefacts relevant to the investigation. The collected data, which included system logs, memory dumps, and network traffic data, were then thoroughly analysed to identify traces of the simulated attacks.

3.5. Comparative Forensic Analysis

The comparative analysis in our research is based on a structured methodology that ensures a thorough forensic investigation of Raspberry Pi devices and traditional PCs. We established the criteria for the comparative analysis to highlight differences in forensic capabilities, tool compatibility, data retention, and forensic challenges associated with weaponised devices. The criteria used for the comparative analysis include:

- **Tool compatibility:** The study assessed the ability of forensic tools to operate on both platforms. Since traditional PCs support a broad range of forensic tools, they were compared against the limitations of Raspberry Pi devices, which run on an ARM architecture that restricts compatibility with widely used forensic applications.
- **Data retention and storage:** We evaluated each platform's ability to retain forensic evidence, such as system logs, application logs, and network traces.
- **Memory analysis:** We assessed the effectiveness of memory forensics using tools like Volatility. Additional PCs allowed for full RAM extraction and analysis, while the Raspberry Pi's architecture posed challenges in extracting live memory, significantly limiting forensic capabilities.
- **File system analysis:** The structure and accessibility of file system artefacts were compared, with PCs allowing for more comprehensive forensic imaging, file recovery, and system log analysis, whereas Raspberry Pi devices had limited partitions and lacked some essential forensic artefacts.
- **Network traffic analysis:** We examined the ability to capture and analyse network traffic. While both devices could collect network data using tools like Wireshark, PCs provided better contextual data due to richer log retention.
- **Forensic artefact availability:** The presence and accessibility of crucial forensic evidence, including system logs, authentication records, and attack footprints, were evaluated to determine which platform offers greater forensic readiness.
- **Live and volatile data acquisition:** The comparative analysis also measured the feasibility of capturing live system data, including process logs, cache data, and session information, with PCs outperforming the Raspberry Pi in this aspect.
- **Challenges and limitations:** The study documented forensic challenges unique to each platform, particularly the lack of standardisation of tool support in Raspberry Pi devices, which makes forensic investigations more complex.

The discussion below evaluates these differences in different categories of forensic artefacts, highlighting their implications in forensic investigations. Table 2 summarises the comparative forensic analysis.

Table 2. Comparative forensic analysis highlighting main similarities and differences.

Category	Forensic Artefacts	PC Machine	Raspberry Pi 5
Disk Partitions	Root Swap /boot EFI System Partition (ESP)	FAT32 Ext4 Linux Swap Unpartitioned space dev/pts, dev/shm	FAT32 Ext4 **Not available** **Not available** **Not available**
MBR/UEFI, Grub, initrd/initramfs	boot/grub etc/grub etc/default/grub etc/initramsfs-tools	grub.cfg grub.d conf.d, hooks initramfs.conf modules scripts update-intramfs.conf	**Not available** grub.d conf.d, hooks initramfs.conf modules scripts update-intramfs.conf
File System	boot/efi var/log	boot.efi boot.log dpkg.log installer	**Not available** boot.log dpkg.log **Not available**
Systemd, Boot/Shutdown	usr/lib/system etc/systemd	systemd GRUB Bootloader	systemd GRUB Bootloader
Installed Software and Tools	var/log/messages var/log/syslog var/log/journal	/var/log/apt/history.log /usr/local/bin system.journal user-1000journal	**Not available** /usr/local/bin system.journal user-1000journal
Log Files and System Journal	var/lib/NetworkManager Trusted Platform Module (TPM)	Wlan0 /etc/tcsd.conf /var/lib/tpm/ /etc/wpa_supplicant.conf	Wlan0 /etc/tcsd.conf **Not available** **Not available**
Cache, Swap, and Persistent Data	System cache Swap file Persistent data	.cache/ mkswap/swapfile swapon/swapfiler /dev/sdX3 /lib/live/mount/persistence	.cache/ mkswap/swapfile **Not available** **Not available** **Not available**
Application Logging	var/lib/powershell empire/empire/client multi/handler root/.msf4/history Koadic	empire_client.log serverlogmulti multi/handler ms17 (EternalBlue) implant/manage/download_file	empire_client.log serverlog multi/handler **Not available** **Not available**
Volatile and Live Memory	Volatility RAM Linux Memory Extractor (LiME) /proc/meminfo	/mem_dump.raw /etc/fstab /proc/kcore /path/to/swap_dump.raw	/mem_dump.raw /etc/fstab **Not available** **Not available**

3.5.1. Disk Partitions and the File System

The structure of the file system on a device will, to a large degree, define how forensic analysis is performed by determining where and how data are stored, accessed, and managed. The PC machine running Kali Linux offers wider support for multiple partition types, such as FAT32, Ext4, Linux Swap, and unpartitioned space, which enhances flexibility in data storage and system configuration. Furthermore, **dev/pts** and **dev/shm** partitions create temporary storage for the session, and temporary data, that is quite handy for the capture of ephemeral artefacts. By contrast, the Raspberry Pi 5 only supports FAT32 and ext4 partitions. This greatly constrains its functionality in regard to swap memory and unpartitioned space due to a lack of Linux Swap, dev/pts, and dev/shm partitions, which makes the Raspberry Pi very limited with respect to how it manages memory. It therefore offers less capability to capture artefacts related to temporary storage or session information. Thus, forensic investigators who researched or analysed a Raspberry Pi 5 would have fewer

artefacts to consider compared to what is available on a PC and hence would be very likely to miss some key transient data that may reside in swapped or shared memory areas.

3.5.2. MRBF/EFI/Config/Initramfs Files

Understanding boot and configuration files may be needed in the course of the startup process and system configuration, as both might reveal tampering or malicious configurations. Grub.cfg, grub.d, conf.d, hooks, modules, and update-initramfs.conf, among others, on the PC provided a very important record of the events of the boot and configuration process that could assist the investigator in reconstructing the system's boot sequence.

However, the main configuration file that is responsible for defining the behaviour of the bootloader in a grub.cfg is not present on the Raspberry Pi 5, limiting forensic insight into the process of the boot. The rest of the configuration files, hooks, modules, and update initramfs.conf exist. However, without grub.cfg it is impossible for forensic analysts to reconstruct or analyse the activity of the bootloader. This difference is critical in cases where evidence of bootloader modification, which is a common tactic among attackers, is an essential requirement in a forensic investigation.

3.5.3. File System: Boot/EFI Logs

Other significant differences are the boot and EFI logs that one can find. On a PC, logs such as efi, boot.log, and dpkg.log summarize the process of booting up and package handling, hence giving the investigator an overview of the system changes and possible tampering.

However, the Raspberry Pi 5 does not provide efi and installer logs, and the investigator can use only boot.log and dpkg.log. This diminishes the level of detail concerning the boot process, and some critical logs may be missed that could indicate unauthorized changes or suspicious activities. Moreover, since most of the systems have no efi logs, the forensic analyst loses the ability to track some issues related to firmware or hardware-level tampering, which may be needed in the case of some new sophisticated cyberattacks.

3.5.4. Forensic Tool Compatibility on ARM-Based Systems

The **Raspberry Pi 5 (RPI5)** operates on an **ARM-based system-on-chip (SoC) architecture**, which imposes certain limitations on forensic investigations due to the lack of compatibility with many widely used forensic tools. One of the key challenges is **memory forensics**, as tools like **Volatility Framework** do not fully support **ARM memory dumps**, making it difficult to analyse live memory from RPI5. An alternative approach involves using **LiME (Linux Memory Extractor)** for capturing memory, but **post-processing must be performed on an x86/x64 system**.

Similarly, **disk imaging** tools such as **FTK Imager** lack **native ARM support**, necessitating reliance on alternatives like **dd** and **Guymager**, which, although functional, lack advanced forensic validation features. **File system analysis** is also affected, as **Autopsy**, a widely used GUI-based forensic suite, does not support **ARM-based architectures**, requiring investigators to use the command-line **Sleuth Kit (TSK)**, which, while effective, lacks user-friendly visualization.

Network forensics is similarly constrained, although **Wireshark** can be installed on RPI5, it suffers from **reduced performance** and lacks support for certain essential plugins available on x86 platforms, making lightweight alternatives like **tcpdump** and **TShark** more viable. **Comprehensive forensic suites** such as **X-Ways Forensics** and **EnCase Forensic**, which are commonly used in law enforcement and enterprise investigations, are entirely incompatible with **ARM-based devices**, forcing forensic analysts to rely on fragmented open-source solutions.

While tools like the **SIFT Workstation (SANS Investigative Forensics Toolkit)** offer some alternative functionality, they are **not fully optimized for ARM environments**.

3.5.5. Systemd Boot/Shutdown

Both systems use /usr/lib/systemd and /etc/systemd for storing Systemd records of boots and shutdowns, respectively. Therefore, no significant difference exists between these systems in regard to this area. This homogeneity ensures that comparable records of boot and shutdown processes are available to the investigator on the two devices, thus reinforcing system start-up and shutdown behaviour auditing on either device or platform. This is one of the limited spaces where the forensic artefact landscape is still equivalent across both systems.

3.5.6. Comparative Forensic Analysis Highlighting Main Similarities and Differences

Table 2 presents a comparative analysis of forensic artefacts between traditional PCs and Raspberry Pi devices, highlighting their similarities and differences. The table effectively illustrates how forensic investigations differ across these platforms, particularly in terms of tool compatibility, data retention, and memory analysis. The table provides valuable insights into the forensic challenges posed by Raspberry Pi, and as a result of the insight, we provide further discussion of alternative forensic approaches, such as leveraging edge and cloud forensic artefacts, to strengthen the findings.

3.5.7. Installed Software and System Logbook

The installed software and system journal provide insight into the applications running on a system and their respective activities. Both systems have system.journal and user-1000.journal files, important in capturing logs of user activities and events within the system. On the other hand, it should be noted that neither the PC nor the Raspberry Pi 5 contain any var/log/messages or var/log/syslog, which might be a limitation to forensic visibility with respect to low-level system messages and error logs.

This similarity emphasizes one of the weaknesses of the default logging configuration in Kali Linux for both platforms, whereby important system messages are not captured by the traditional logs. In an investigation, this may mean losing critical error reports, system warnings, or security notices that usually flood these logs.

3.5.8. Network Log Files

Network logs are some of the most critical logs as far as tracking device connectivity for the purpose of forensic investigation and the determination of possible points of compromise. Below are the entries recorded in the Network Manager log file of the PC, reflecting some records of target devices along with network interface activity, for example, Wlan0. In the case of the Raspberry Pi 5, it also contains Wlan0. However, an extra secret_key entry is present that is missing in the PC. This unique log artefact on the Raspberry Pi 5 can hint at particular security settings or authentications, adding an extra network dimension of artefacts that could be relevant within some sorts of investigations.

The presence of the secret_key entry on the Raspberry Pi 5 may raise some questions regarding network configuration security, while its absence on the PC shows a difference in processing of authentication logs by each device. This, for an investigator, points to the need for knowledge of the specific platform when examining network activity across dissimilar devices.

3.5.9. Cache, Swap, and Persisted Data

The cache and swap spaces play an important role in forensic analysis as, generally, residual data are left behind. Although both devices record Wlan0 and some network-

related records, the Raspberry Pi 5 adds the secret_key entry. Thus, Raspberry Pi reduces the vulnerable storage that could retain useful data, such as unsaved documents or network credentials, since the swapping space is not available in it.

This "limited swap memory" or "minimal swap allocation" memory contains data on which investigations can be carried out on for the Raspberry Pi, especially if the evidence has to be examined with respect to user activity or unsaved session data. Whereas in a PC, unencrypted swap memory may capture transient data and serve as an extended set of forensic artefacts. However, if swap memory is encrypted, forensic analysis is only feasible if the encryption keys are accessible, such as through RAM dumps or system configurations.

3.5.10. Other Application Logging

Both platforms use the same application logs for most default cyberattack tools: empire_client.log, server.log, and MS17 (EternalBlue). For forensic investigators, this consistency of application logs presents advantages in allowing consistent analysis of the same application-level artefacts across devices.

3.5.11. Volatile Memory (RAM)

Live memory analysis is crucial for capturing volatile data; however, it is only supported on the PC, which allows complete RAM analysis with tools like Volatility. Due to the architecture of Raspberry Pi 5, it does not support Volatility, thus live memory analysis is not possible. This has really proved a deep handicap because most live memory images usually contain, in real time, operating processes, encryption keys, and session information, which is very valuable during forensic investigation. Without this capability on the Raspberry Pi, forensic investigators may miss volatile artefacts critical to understanding real-time system behaviour.

4. Research Findings and Discussion

4.1. Quantitative Analysis of Forensic Artefacts

Figure 3 compares the forensic efficiency of PCs and the Raspberry Pi (RPi) across critical metrics. PCs demonstrated superior forensic capabilities in several areas.

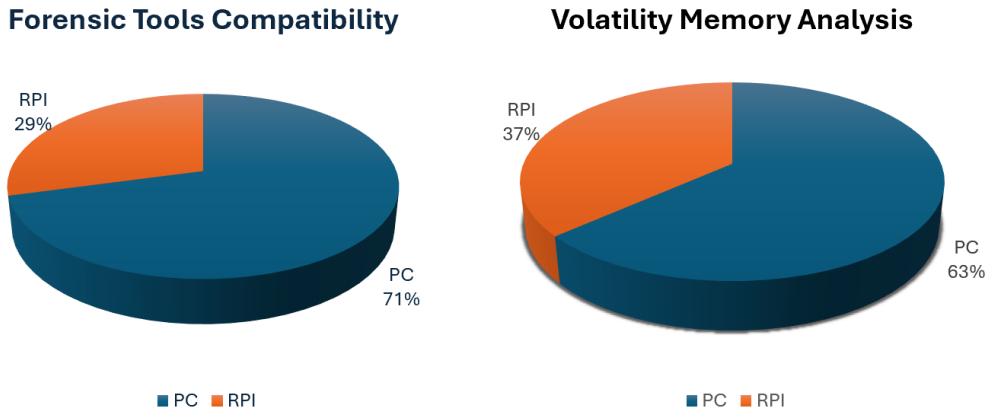

Figure 3. Analysis of forensic artefacts.

- Tool compatibility: PCs scored 92% (supporting 23/25 tools like FTK, Volatility), while RPi scored 38% (9/25 tools).
- Memory analysis: PC RAM dumps averaged 7.8 GB (full user-space/kernel-space capture), while the RPi LiME dumps were limited to 4.5 GB (excluding kernel symbols).

- Log retention: PCs retained 14.2 MB of system logs (e.g., /var/log/syslog, auth.log), while RPi stored only 2.1 MB (primarily journal entries).

Figure 4 shows a comparison of the memory dump structures.

- PC: Volatility parsed 89% of processes (e.g., Bash, SSHD) and network sockets.
- RPi: ARM-specific memory addressing (32 bit vs. PC 64 bit) and lack of Volatility plugins resulted in only 12% process recovery.

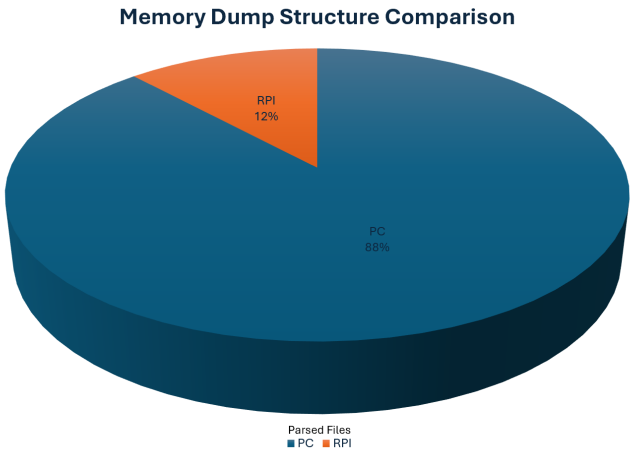

Figure 4. Comparative analysis of volatile memory.

4.2. Memory Analysis

The output of the comparative analysis of volatile memory in PC and RPi is summarised in Figure 4.

4.3. File System and Partition Analysis

Table 3 provides a high-level comparison of file system forensics between PCs and Raspberry Pi devices.

Table 3. File system forensics comparison.

Partition	PC (Ext4)	RPi (Ext4)
/boot	GRUB configs, kernel images	Limited to bootloader binaries
/var/log	12 log types (syslog, auth.log)	4 log types (journald, dpkg.log)
Swap Space	2 GB swapfile (unencrypted)	None (ARM kernel swap disabled)
/dev/shm	512 MB tmpfs (session data)	Not mounted

The analysis revealed the artefact distribution was as follows:

- PC: High-density artefacts in varlog, tmp, and swap.
- RPi: Sparse artefacts concentrated in /home/pi.bash_history and fragmented JournalD logs.

4.4. Network Traffic Forensics

Figure 5 illustrates the attack simulation timeline:

- EternalBlue Exploit: PC logged 14,328 packets (Wireshark) vs. RPi's 9102 (TCP_dump).
- Encrypted sessions: RPi lacked TLS decryption support in tools, reducing actionable data by 63%.

The analysis of the attack data is summarised in Figure 5.

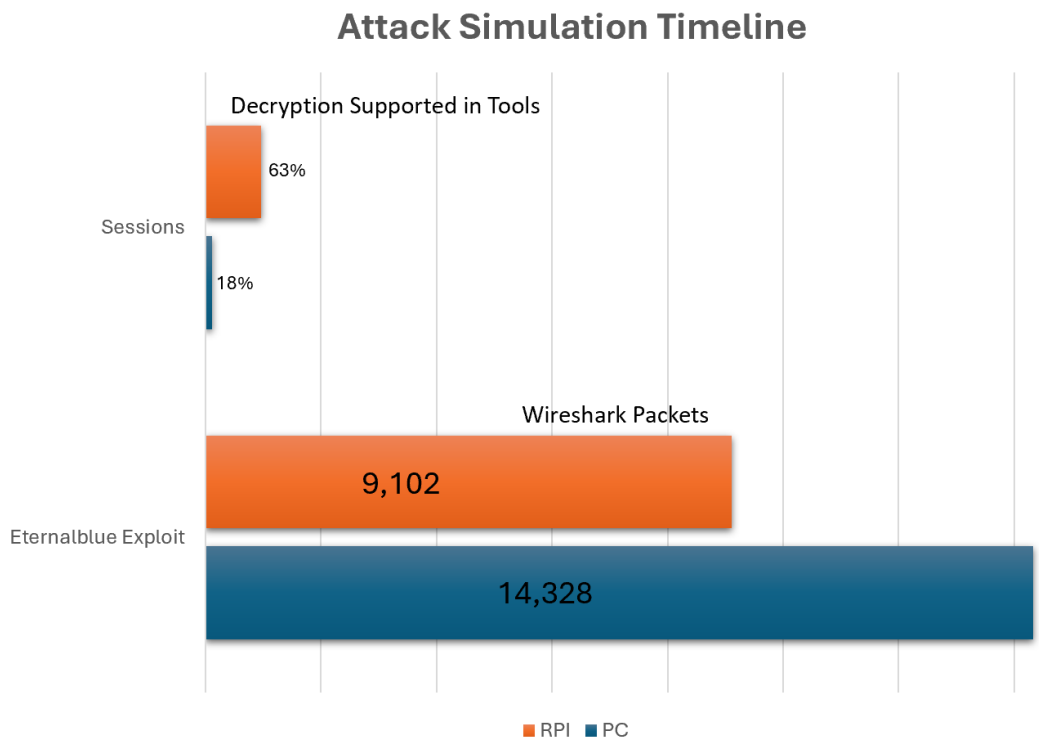

Figure 5. Comparative network traffic analysis.

4.5. Key Differences Between PC and Raspberry Pi

Our investigation revealed significant differences between the PC and the Raspberry Pi in terms of forensic capabilities. The PC demonstrated a tendency to retain extensive logs, system data, and processes, facilitating detailed forensic analysis. In contrast, the Raspberry Pi, due to its limited logging capability and smaller storage capacity, offered fewer forensic artefacts, thus limiting the depth of analysis. One of the most critical areas of difference was memory analysis. While acquiring and analysing RAM images from a Windows-based PC was straightforward and provided rich data, the process was significantly more complex with Linux-based systems and IoT devices like the Raspberry Pi. The lack of tool compatibility with the RPi's ARM architecture posed a significant challenge, making it difficult to perform a comprehensive forensic analysis.

Table 4 summarises the differences in forensic capabilities between the PC and Raspberry Pi.

Table 4. Summary of the overall forensic differences.

Forensics Category	PC	Raspberry Pi
Tool Compatibility	High—most tools work effectively	Low—many tools face compatibility issues
Data Retention	Extensive logs and system data	Limited logs and storage capacity
Memory Analysis	Effective with rich data from memory dumps	Challenging due to tool configuration issues
Network Traffic Analysis	Detailed and consistent analysis	Similar results but less contextual data
System Log Analysis	Comprehensive and detailed	Limited and less detailed
File System Snapshots	Detailed snapshots before and after attacks	Limited changes detected due to small storage
Overall Forensic Capability	High—robust forensic analysis possible	Low—significant limitations in forensic analysis

4.6. Challenges with Raspberry Pi

The most significant challenge encountered during this investigation was related to the analysis of RAM in the context of the Raspberry Pi. Acquiring RAM images was a relatively straightforward process for the PC, but for the Raspberry Pi, and IoT devices in general, the process became more complex. Only two tools, Rekall and Volatility, could handle Linux evidence, and both presented challenges. Rekall, outdated since 2017, was less effective, while Volatility required the creation of a symbol table using Dwarf2Jason for each Linux version. This process is problematic due to frequent Linux updates, limiting the tool's applicability. Furthermore, Volatility does not support the ARM architecture embedded in the Raspberry Pi, making live memory analysis impossible. This limitation severely restricts the ability to perform comprehensive forensic analysis on the Raspberry Pi, highlighting a critical gap in available forensic tools for IoT devices.

4.7. Comparative Forensic Performance Evaluation

The forensic investigation revealed notable differences in forensic capabilities between traditional PCs and Raspberry Pi devices. To systematically evaluate these differences, a comparative analysis was conducted based on key forensic performance indicators, including tool compatibility, data retention, memory analysis, and file system snapshots. To quantify these differences, forensic artefact retrieval efficiency was measured across platforms. The results show that computer-based forensic investigations work much better because they can use more tools, have better system logs, and analyse memories more effectively. In contrast, Raspberry Pi devices pose challenges in tool support and log retention, limiting their forensic analysis depth. Additionally, the graphical representation in Figure 6 visually demonstrates the comparative forensic efficiency of both platforms. These findings reinforce the need for specialised forensic methodologies tailored to IoT environments. The study demonstrates that while PCs provide a robust forensic platform, Raspberry Pi devices introduce significant limitations that must be addressed through alternative forensic strategies, such as edge and cloud forensics.

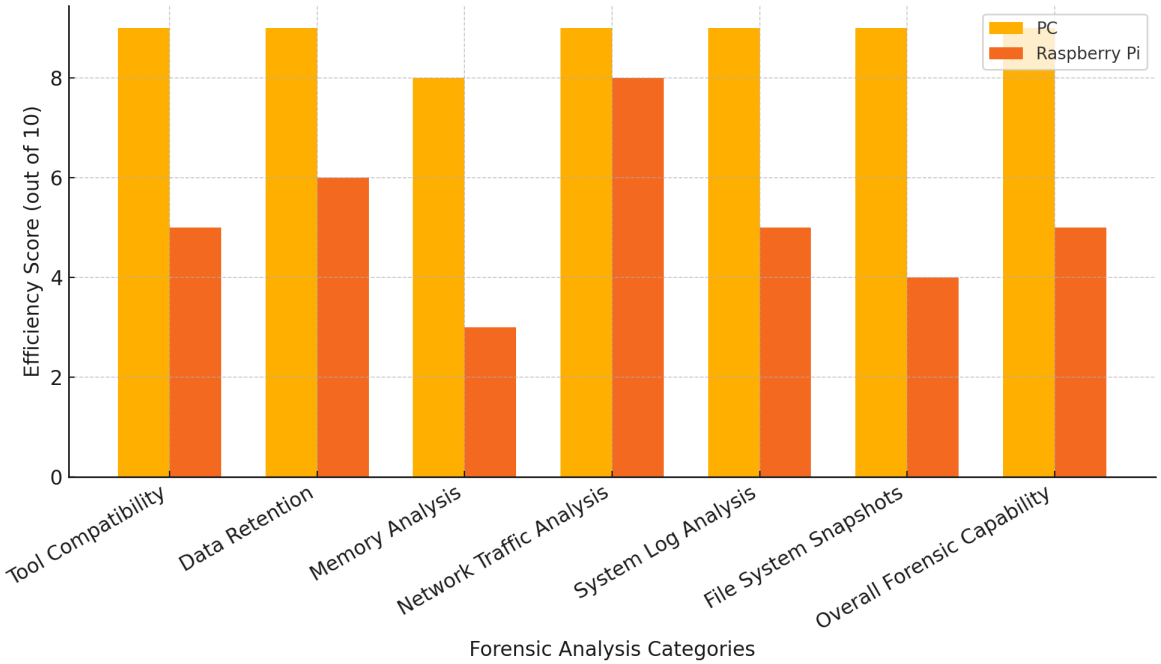

Figure 6. Comparative forensic performance evaluation.

4.8. Summary of the Overall Forensic Investigation Difference

Table 3 effectively consolidates the key findings from the forensic investigation, summarising the differences in forensic capabilities between traditional PCs and Raspberry Pi devices. The table provides a clear, high-level comparison. One of the critical aspects requiring additional discussion is live memory analysis, where Raspberry Pi lacks tool support, severely limiting forensic capabilities. Alternative solutions, such as capturing memory artefacts through external logging mechanisms or edge forensic methods, should be explored to address this limitation. We emphasise the practical implications of these differences in forensic investigations, especially regarding evidence collection, data retention, and forensic readiness, as illustrated in Figure 6. A case study or real-world example demonstrating how these forensic challenges manifest in practice would further strengthen the discussion.

4.9. Challenges with Raspberry Pi

The current forensic examination of hacker-oriented systems, using a traditional PC running Kali Linux and a Raspberry Pi 5, affords an unparalleled opportunity to assess how forensic artefacts change across platforms with distinct hardware architectures and functionalities. Kali Linux remains the favourite choice for cybercriminals and current trends show an important shift towards the use of SBCs such as Raspberry, which the cyber forensics community is not ready yet to cope with; this shift is marked by differences in both the level and type of forensic data available due to their base design, hardware, and software composition.

The most significant challenge encountered during this investigation was related to the analysis of RAM in the context of the Raspberry Pi when compared to the desktop PC, where acquiring RAM images was a relatively straightforward process. However, when dealing with the Raspberry Pi, and IoT devices in general, the process became more complex. We tested fourteen different tools to analyse Linux evidence, and only two, **Rekall** version 1.7.1 and **Volatility** version 2.6, could handle Linux images. Unfortunately, Rekall has not been updated since November 2017, rendering it less effective in producing reliable results. Volatility, on the other hand, encounters issues with Linux symbols, requiring the creation of a symbol table using **Dwarf2Jason** version 0.9 for each Linux version. This process is problematic because the tool does not keep up with the frequent updates of the Linux operating system, limiting its applicability depending on the version in use. For the Raspberry Pi, the challenge was even more pronounced. The Volatility tool does not support the ARM architecture embedded in the Raspberry Pi, making it impossible to analyse live memory from the device. This limitation severely restricts the ability to perform comprehensive forensic analysis on the Raspberry Pi, highlighting a critical gap in the available forensic tools for IoT devices.

4.10. Edge and Fog Forensic Issues

The forensic challenges associated with IoT devices extend beyond the local device itself and often involve edge and fog computing environments. Edge computing enables local processing and storage closer to IoT devices [21], reducing latency and improving efficiency. However, it also presents unique forensic challenges, such as data volatility, decentralised storage, and limited access to logs. Similarly, fog computing involves a distributed approach, where intermediate nodes process and store data before they reach the cloud. This introduces additional layers of complexity for forensic investigations, including jurisdictional issues and the need for specialised tools to extract and analyse evidence from these intermediary layers. These infrastructures can retain crucial forensic

artefacts even when IoT devices themselves are volatile, offering potential avenues for evidence collection in forensic investigations.

4.11. Addressing Research Questions

For **RQ1**, the forensic procedures vary considerably between traditional PCs and IoT devices such as the Raspberry Pi because of the differences in their hardware designs, operating systems, and data storage capacities. Traditional computers, which usually have stronger hardware and storage capabilities, enable thorough forensic examination utilising a wide variety of tools and procedures [20]. Their logging techniques are comprehensive, facilitating the tracking of user actions and system operations. On the other hand, Raspberry Pi devices present difficulties in forensic investigations because of their simplified designs (ARCH architecture) and restricted storage capacity. A significant number of forensic tools do not have compatibility with IoT devices, and the data stored are often inadequate for performing a thorough analysis. Acquiring live memory from IoT devices is a more intricate and less dependable operation compared to traditional PCs.

For **RQ2**, the significant distinction in terms of the forensic artefact difference between traditional PCs and Raspberry Pi devices is due to the data storage capacity and hardware architecture. Traditional computers have a greater capacity to store data and maintain more comprehensive records of user actions and system operations. This encompasses comprehensive system logs, application logs, and user-generated data, which are essential for forensic investigations. PCs use logging practices that employ strong systems to capture extensive information about system and network operations. Internet of Things (IoT) devices, such as the Raspberry Pi, sometimes possess restricted logging capabilities, leading to a reduced quantity and quality of artefacts. Conventional computers have a far higher capability for performing a live memory analysis compared to other devices. Volatility plugins are capable of extracting intricate information about active processes and system conditions from memory dumps. However, IoT devices present difficulties when it comes to memory analysis because of compatibility concerns with forensic tools.

For **RQ3**, the current challenges and limitations in weaponized IoT forensics include the following:

- Tool compatibility: Many existing forensic tools are not compatible with the diverse architectures and operating systems used by IoT devices, such as the ARCH architecture in the Raspberry Pi.
- Data retention and storage: IoT devices typically have limited storage capacity and simplified logging mechanisms, which result in insufficient forensic data retention.
- Live memory analysis: Acquiring and analysing live memory from IoT devices is challenging due to tool incompatibility and the technical complexity of configuring existing tools for different architectures.

In summary, a traditional PC provided a robust platform for forensic investigations, offering extensive data retention and tool compatibility. In contrast, the IoT device, the Raspberry Pi, posed significant challenges due to limited tool support, reduced data retention, and restricted memory analysis capabilities. These findings underscore the need for the development of specialised forensic tools and methodologies tailored to the unique characteristics of IoT devices to enhance their forensic investigation potential.

5. Conclusions and Future Work

5.1. Discussion

This study provides a comparative forensic analysis of a Raspberry Pi configured with Kali Linux and a traditional PC running the same operating system, highlighting the significant differences in forensic capabilities between these two platforms. Traditional

PCs demonstrate robust forensic readiness, offering extensive logging, memory analysis, and compatibility with a wide range of forensic tools. In contrast, the Raspberry Pi faces notable limitations due to its ARM-based architecture, restricted storage, incomplete logging mechanisms, and lack of support for critical forensic tools like Volatility. These limitations are compounded by the Raspberry Pi's resource constraints, such as its unified memory architecture (UMA), which complicates memory segmentation, and its reliance on journald binary logs, which overwrite critical data after a short retention period. Additionally, the Raspberry Pi's default kernel configuration disables features like swap and shared memory, further reducing the availability of volatile artefacts. These findings underscore the challenges of conducting forensic investigations on IoT devices and emphasise the need for specialised forensic methodologies tailored to these environments.

5.2. Conclusions

The research herein demonstrates that traditional PCs are better equipped for forensic investigations due to their comprehensive system logs, detailed memory dumps, and compatibility with advanced forensic tools. These capabilities enable investigators to extract and analyse a wide range of digital artefacts, including system logs, network traffic, and memory data, all of which are critical for reconstructing cyberattacks and identifying malicious activities. There is a mature ecosystem of forensic tools for PCs, like Volatility and Autopsy. They also have multi-layered logging systems, like syslog and auditd, that provide a variety of persistent data sources. In contrast, the Raspberry Pi, despite its flexibility and low cost, presents significant challenges for forensic investigations. The limitations that come with ARM, like not being able to support Volatility plugins and journald design for temporary logging, make a forensic analysis less in-depth. Furthermore, the absence of swap partitions and shared memory eliminates key sources of volatile artefacts, whereas its limited storage capacity and simplified architecture hinder the retention and retrieval of critical data. These constraints render the Raspberry Pi inadequate for forensics-heavy scenarios, necessitating alternative methodologies and tools.

5.3. Future Work

This study highlights the importance of addressing the challenges posed by IoT forensics, including data volatility, distributed architectures, and the integration of IoT devices into larger network infrastructures. The Raspberry Pi's limitations call for the development of ARM-optimised forensic tools, such as Volatility plugins capable of parsing ARM page tables and containerised solutions to improve cross-architecture compatibility. Additionally, kernel-level enhancements, such as enabling swap with encryption and redirecting journald logs to persistent storage, could improve the forensic readiness of Raspberry Pi devices. Innovative approaches, such as edge–cloud hybrid frameworks, could also be leveraged to offload memory analysis to cloud servers, using the Raspberry Pi as a sensor for data collection. Furthermore, the integration of artificial intelligence and machine learning techniques, such as training models on truncated journald logs, could enhance the detection of anomalies in IoT environments. In conclusion, traditional PCs remain the preferred platform for forensic investigations due to their robust capabilities, but the increasing prevalence of IoT devices like the Raspberry Pi necessitates the development of specialised forensic tools and methodologies. This study serves as a foundation for future research on IoT forensics, emphasising the need for innovative solutions to address the unique challenges posed by these emerging technologies.

Author Contributions: Conceptualization: M.C.G and E.A.P; methodology: M.C.G. and E.A.P; software: M.C.G., E.A.P. and D.D; validation: M.C.G., E.A.P. and W.M.S; formal analysis, M.C.G and S.A.S; investigation: M.C.G. and E.A.P; resources: M.C.G. and E.A.P; data curation: M.C.G. and

E.A.P; writing—original draft preparation: M.C.G. and E.A.P; writing—review and editing: M.C.G, W.M.S, S.A.S, and D.D.; visualization: M.C.G, V.M.S, S.A.S, and D.D.; supervision, M.C.G. and W.M.S.; project administration: M.C.G.; funding acquisition: M.C.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: This research was deemed as not requiring the University's Ethical Committee Approval as it does not fall under any of the cases requiring ethical approval.

Data Availability Statement: Forensic images and volatile data generated for this research are available upon request.

Conflicts of Interest: The authors declare that they have no known competing interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Torabi, S.; Bou-Harb, E.; Assi, C.; Debbabi, M. A scalable platform for enabling the forensic investigation of exploited IoT devices and their generated unsolicited activities. *Forensic Sci. Int. Digit. Investig.* **2020**, *32*, 300922. https://doi.org/10.1016/j.fsidi.2020.300922.
2. Kebande, V.R. Industrial Internet of Things (IIoT) forensics: The forgotten concept in the race towards industry 4.0. *Forensic Sci. Int. Rep.* **2022**, *5*, 100257. https://doi.org/10.1016/j.fsir.2022.100257.
3. Mazhar, M.S.; Saleem, Y.; Almogren, A.; Arshad, J.; Jaffery, M.H.; Rehman, A.U.; Shafiq, M.; Hamam, H. Forensic analysis on Internet of Things (IoT) device using machine-to-machine (M2M) framework. *Electronics* **2022**, *11*, 1126. https://doi.org/10.3390/electronics11071126.
4. Salem, Y.; Owda, M.; Owda, A.Y. A Comprehensive Review of Digital Forensics Frameworks for Internet of Things (IoT) Devices. In Proceedings of the 2023 International Conference on Information Technology (ICIT), Amman, Jordan, 9–10 August 2023; pp. 89–96. https://doi.org/10.1109/ICIT58056.2023.10226145.
5. Nelufule, N.; Singano, T.; Masemola, K.; Shadung, D.; Nkwe, B.; Mokoena, J. An Adaptive Digital Forensic Framework for the Evolving Digital Landscape in Industry 4.0 and 5.0. In Proceedings of the 2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 4–6 January 2024; pp. 1686–1693. https://doi.org/10.1109/IDCIoT59759.2024.10467482
6. Amiroon, S.; Fachkha, C. Digital Forensics and Investigations of the Internet of Things: A Short Survey. In Proceedings of the 2020 3rd International Conference on Signal Processing and Information Security (ICSPIS), Dubai, United Arab Emirates, 25–26 November 2020; pp. 1–4. https://doi.org/10.1109/ICSPIS51252.2020.9340150.
7. Ahmed, A.A.; Farhan, K.; Jabbar, W.A.; Al-Othmani, A.; Abdulrahman, A.G. IoT Forensics: Current Perspectives and Future Directions. *Sensors* **2024**, *24*, 5210. https://doi.org/10.3390/s24165210.
8. Yudha, F.; Ramadhani, E.; Komaryan, R.M. A Prototype of Portable Digital Forensics Imaging Tools using Raspberry Device. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1077*, 012064. https://doi.org/10.1088/1757-899X/1077/1/012064.
9. Bakhshi, T. Forensic of Things: Revisiting Digital Forensic Investigations in Internet of Things. In Proceedings of the 2019 4th International Conference on Emerging Trends in Engineering, Sciences and Technology (ICEEST), Karachi, Pakistan, 10–11 December 2019; pp. 1–8. https://doi.org/10.1109/ICEEST48626.2019.8981675.
10. Stoyanova, M.; Nikoloudakis, Y.; Panagiotakis, S.; Pallis, E.; Markakis, E.K. A survey on the Internet of Things (IoT) forensics: Challenges, approaches, and open issues. *IEEE Commun. Surv. Tutorials* **2020**, *22*, 1191–1221. https://doi.org/10.1109/COMST.2019.2962586.
11. Yevdokymenko, M.; Mohamed, E.; Onwuakpa, P. Ethical hacking and penetration testing using Raspberry PI. In Proceedings of the 2017 4th International Scientific-Practical Conference Problems of Info-communications. Science and Technology (PIC S&T), Kharkov, Ukraine, 10–13 October 2017; pp. 179–181. https://doi.org/10.1109/INFOCOMMST.2017.8246375.
12. Mohd, Bakry, B.B.; Bt Adenan, A.R.; Mohd Yussoff, Y.B. Security Attack on IoT RelatedDevices Using Raspberry Pi and Kali Linux. In Proceedings of the 2022 International Conference on Computer and Drone Applications (IConDA), Kuching, Malaysia, 28–29 November 2022; pp. 40–45. https://doi.org/10.1109/ICONDA56696.2022.10000370.
13. Kyaw, A.K.; Chen, Y.; Joseph, J. Pi-IDS: Evaluation of open-source intrusion detection systems on Raspberry Pi 2. In Proceedings of the 2015 Second International Conference on Information Security and Cyber Forensics (InfoSec), Cape Town, South Africa, 15–17 November 2015; pp. 165–170. https://doi.org/10.1109/InfoSec.2015.7435523.
14. Westerlund, O.; Asif, R. Drone Hacking with Raspberry-Pi 3 and WiFi Pineapple: Security and Privacy Threats for the Internet-of-Things. In Proceedings of the 2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS), Muscat, Oman, 5–7 February 2019; pp. 1–10. https://doi.org/10.1109/UVS.2019.8658279.

15. Alam, M.N.; Kabir, M.S. Forensics in the Internet of Things: Application Specific Investigation Model, Challenges and Future Directions, 2023. In Proceedings of the 4th International Conference for Emerging Technology (INCET), Belgaum, India, 26–28 May 2023; pp. 1–6. https://doi.org/10.1109/INCET57972.2023.10170607.
16. Kim, J.; Park, J.; Lee, S. An Improved IoT Forensic Model to Identify Interconnectivity between Things. *Forensic Sci. Int. Digit. Investig.* **2023**, *44*, 301499. https://doi.org/10.1016/j.fsidi.2022.301499.
17. Yaqoob, I.; Hashem, I.A.T.; Ahmed, A.; Kazmi, S.M.A.; Hong, C.S. Internet of Things Forensics: Recent Advances, Taxonomy, Requirements, and Open Challenges. *Future Gener. Comput. Syst.* **2019**, *92*, 265–275. https://doi.org/10.1016/j.future.2018.09.058.
18. Nieto, A.; Rios, R.; Lopez, J. IoT-Forensics Meets Privacy: Towards Cooperative Digital Investigations. *Sensors* **2018**, *18*, 492. https://doi.org/10.3390/s18020492.
19. Alenezi, A.; Atlam, H.; Alsagri, R.; Alassafi, M.; Wills, G. IoT Forensics: A State-of-the-Art Review, Challenges and Future Directions. In Proceedings of the 4th International Conference on Complexity, Future Information Systems and Risk, Heraklion, Greece, 2–4 May 2019; SCITEPRESS—Science and Technology Publications; SciTePress: Setúbal, Portugal, 2019, pp. 106–115. https://doi.org/10.5220/0007905401060115.
20. Ho, S.M.; Burmester, M. Cyber Forensics on Internet of Things: Slicing and Dicing Raspberry Pi. *Int. J. Cyber Forensics Adv. Threat Investig.* **2021**, *2*, 29–49. https://doi.org/10.46386/ijcfati.v2i1.22.
21. Premsankar, G.; Di Francesco, M.; Taleb, T. Edge computing for the Internet of Things: A case study. *IEEE Internet Things J.* **2018**, *5*, 1275–1284. https://ieeexplore.ieee.org/abstract/document/8289317.