International Journal of Information Security manuscript No.
(will be inserted by the editor)

Integrating the Edge Computing Paradigm Into the Development of IoT Forensic Methodologies

Juan Manuel Castelo Gómez · Sergio Ruiz-Villafranca

Received: date / Accepted: date

Abstract With the number of attacks and cyberincidents affecting Internet of Things (IoT) devices on the rise, the need for carrying out forensic investigations to determine what has happened has grown at the same pace. However, due to the characteristics and requirements of this environment, the solutions used until now in the field are not suitable to be followed, as they are not able to guarantee the effective retrieval and study of the pieces of evidence. Under these circumstances, new ways of interacting with IoT units are needed, and the edge computing paradigm has emerged as an interesting asset to complement IoT networks. Attack, anomaly and intrusion detection, or data encryption are some of the fields in which this approach has been successfully applied. Following this concept, this article presents an IoT forensic methodology that integrates the edge computing technology in order to assist in the investigation process, trying to address some of the issues that are hindering the effectiveness of examinations in this scenario. In addition, this proposal is compared with the existing ones in the research community, and evaluated by testing it in two case studies representing real-life scenarios, ultimately demonstrating suitability to be used for IoT forensic examinations.

Keywords Cybersecurity · IoT · IoT forensics · Edge Computing · Forensic Methodology

Juan Manuel Castelo Gómez
Technical University of Madrid. Alan Turing s/n. 28031. Madrid (Spain). E-mail: juanmanuel.castelo@upm.es. Corresponding author

Sergio Ruiz-Villafranca
University of Castilla-La Mancha. Avda. de España s/n. 02071. Albacete (Spain). E-mail: sergio.rvillafranca@uclm.es

1 Introduction

Carrying out forensic investigations on the Internet of Things (IoT) is no longer far-fetched thought. After seeing the growth that this environment has had in the last few years, with the number of devices surpassing the 13 billion in 2022 [30], and the consequences that this has had in terms of attacks, the IoT has become a source of incidents that, ultimately, need to be studied in order to determine what has happened, thus requiring the opening of a forensic process.

As it occurred when new digital environments appeared, the forensics field is still studying the requirements and characteristics that the IoT has so that solutions can be developed accordingly, and be able to guarantee complete and efficient investigations. However, until these IoT-centered tools arrive, investigators see themselves forced to rely on conventional ones. This aspect limits the effectiveness of their investigations, as these tools are not designed with IoT forensic requirements in mind.

To this end, there are three IoT features that have a crucial impact on forensic examinations: the number of devices in a network, its dynamism, and the lifetime of the evidence. The first two aspects cause an increase in the range of an investigation, as there is a higher number of devices to analyze compared to traditional ones. In addition, there are scenarios in which IoT devices can enter and exit an IoT network in a short period of time, such as vehicular networks. This means that a possible source of evidence may not be present when an investigator starts their examination, as well as makes it quite difficult to keep track of the elements that have generated or are generating data in an IoT network.

With respect to the lifetime of the evidence, IoT devices are designed to be constantly exchanging data

and operate together. Consequently, most of the data that is generated in a network is done in the form of network packets. The disadvantage from the forensic perspective is that their data is lost once that the packet is received.

Furthermore, due to the interoperability of the environment, IoT devices are designed accordingly, making them compatible with several network protocols, but limiting their storage, which means that even fewer data would be found in the non-volatile memory, the largest source of data in conventional scenarios, and the one which has the largest lifetime, and more will be exchanged on-the-fly.

On top of that, not all the conventional acquisition techniques for non-volatile memory are feasible in this environment. When it comes to carrying out physical ones, namely extraction and acquisition, Joint Test Action Group (JTAG), In-System Programming (ISP), or chip-off, investigators find that not all of them can be carried out [46,8,28]. The most logical and immediate reason being that physical access to the device is mandatory, and in an environment in which units are embedded into machines or separated by miles, it is something that is not always a guarantee. Regarding the techniques, the simple extraction and acquisition, quite common when working with computer storage, cannot be performed due to almost all IoT devices having its storage soldered to its board. JTAG, ISP and chip-off are not always compatible and are not easy to perform, as they require of specific equipment and soldering skills. Furthermore, the chances of damaging the IoT unit are quite high when performing them, given that the techniques themselves are harmful, and that they require access to the device's board, so a disassembly is needed. All these issues can be solved by following a remote acquisition approach, but this method has not been perfected over the years, as it was rarely used in conventional scenarios [51].

Even more worrisome is the scenario for the acquisition of volatile memory, which is almost impossible to collect. Firstly, it is necessary to establish a remote connection with the device, so that the acquisition tool can be executed. This is not always possible since not all IoT devices use an operating system and, even if they do, this option is not always enabled. Secondly, memory acquisition tools need to load a kernel module on Linux based systems (the most used by IoT devices) and to create a profile of the volatile memory so that it can be examined. These two processes are not feasible at all on IoT devices, so only carrying out a remote analysis of the device will allow the extraction of information from the volatile data [27,14,10].

The silver lining appears when focusing on the last type of evidence that can be found in a device, namely network traffic. In this case, the procedure of acquiring the packets exchanged in the network is entirely feasible, as it can be done from an external device. This aspect, added to the above-mentioned importance that these data have, makes network traffic an interesting source of evidence in IoT investigations.

With this in mind, a new way of interacting with the sources of evidence is needed so that, at least, the network traffic can be captured when relevant data are being exchanged, which is usually just when the incident arises. In order to do this, it is necessary for the capturing device to be inside the network from which the traffic is going to be acquired. Taking these two aspects into account, an interesting solution would be to insert a device into the IoT network that the investigator wants to examine, and then collect the necessary data. This way, the investigator ensures that they would be able to examine some data, which would not be possible in other scenarios due to the issues mentioned above.

Under these circumstances, the use of the edge paradigm emerges as a very interesting approach, as it fulfils the listed requirements. It operates the closest possible to the source of data, and does so without interfering in the behaviour of the network. In addition, edge devices are capable of executing more complex tasks than ordinary IoT devices, as the former are more powerful. Its effectiveness has already been confirmed when used in other aspects of IoT cybersecurity, especially on attack and intrusion detection [9,21,47]. Therefore, it seems reasonable to think that its application on IoT forensic would also be of interest for the field.

1.1 Research Questions

With the goal of this research set, the following questions arise when evaluating its feasibility:

- **(RQ1)** Given the lack of IoT-centered forensic tools, in which aspects, if any, can edge devices be useful to assist investigators in IoT examinations?
- **(RQ2)** Seeing the limitations that IoT models, methodologies, and frameworks have in terms of usability in a court of law as a result of their novelty, can the approach of adapting conventional solutions to the IoT's requirements meet the expected criteria when it comes to handling and preserving pieces of evidence?
- **(RQ3)** Considering the multiple approaches followed by the forensic community when it comes to identifying sources of evidence, such as network zone

division, relying on logical communications, or using external solutions, in which aspects using an edge node could be an interesting way of tackling this phase?
- **(RQ4)** Performing the acquisition of IoT devices has proven to be a difficult task due to the impossibility, in many cases, of carrying out either a physical or remote technique. Taking into account that there are no IoT-centered procedures to collect the data generated by IoT devices, how can an investigator overcome this issue and therefore be able to perform an analysis of the devices under examination?

1.2 Contributions

The main contributions of this study are the following:

- We present a review of the proposals from the research community that focus on the design of forensic procedures for the IoT. In addition, we study how the edge computing paradigm is being used in the cybersecurity field in order to evaluate its applicability in forensics.
- We propose an IoT forensic methodology that integrates the edge computing paradigm into the development of IoT forensic solutions. This is done in the form of an edge node that assists in some phases of the investigation process, addressing some of the issues that currently hinder examinations in this environment. By interacting with this node, the investigator can perform several forensic tasks, which results in a reformulation of some of the phases that are carried out in the forensic process.
- We integrate aspects from conventional investigations together with the requirements that the IoT has when it comes to performing forensic examination in it in order to design our proposal. This way, we try to ensure that the standards expected by the community in terms of evidence handling and preservation are met, which means that the proposal can be used in a court of law, as well as guarantee that the methodology is complete and effective enough to be used in the IoT environment.
- We automatize and assist key processes of a forensic investigation, such as the identification and acquisition phases. This eases the process of determining the range of the investigation, the devices that are (and have been), present in the scene, their behaviour, and whether it is possible to carry out a remote acquisition of them. In addition, we design the methodology so that it can be used in a proactive approach, thus providing the IoT network with a higher degree of forensic-readiness than in ordinary scenarios.
- We compare the proposal with the related work that present IoT methodologies, models, frameworks, or guidelines that use any kind of service, piece of software, or hardware to carry out or assist in the investigation process. As a result, we show that the presented methodology improves the existing ones in practicality, level of detail, and feasibility. Furthermore, its scope is the whole IoT environment, not a specific context of it.
- We also evaluate the performance of the proposal when used as a guideline to be followed in IoT forensic investigations in two practical case studies. The scenarios presented are two main contexts in which IoT devices can be found, namely the smart home and the Industrial Internet of Things (IIoT). The case studies demonstrate that the presented methodology is an interesting solution to use both in a proactive and a reactive way when it comes to performing IoT examinations.

The rest of the paper is organized as follows. Section 2 discusses the proposals from the research community regarding IoT forensic methodologies and the use of edge computing in this environment, and compares them with our mehtodology. The proposed solution that integrates the edge technology is detailed in Section 3. Its practical evaluation is presented in Section 4. Section 5 present the answers to the research questions formulated in this introduction. Finally, the conclusions that can be drawn from this research are described in Section 6.

2 Background

In this section, a study of the proposals from the community is presented. Firstly, the IoT methodologies, models, or frameworks that use any kind of solution, either hardware or software-based, to assist in the forensic process are reviewed. Secondly, an examination of the pieces of research which describe the impact that the edge paradigm has on the IoT is presented. After studying these two fields of research, a comparison is presented describing how this proposal differs from the existing work. Finally, the model which is used as a reference for developing the current proposal is described, highlighting the conceptual differences between them.

2.1 IoT Forensic Procedures

Using the first approach to an IoT methodology as a reference [34], [36] presents a methodology which relies

on an external Hadoop server for covering the whole investigation process. The reason behind it is helping on storing the large quantity of data that can be found on IoT examinations. It mentions useful aspects such as the need of issuing warrants to access the data, triage examination and the chain of custody, although it fails to provide a reasonable degree of detail, and no instructions are given on how to perform the tasks, limiting to just narrating an IoT investigation without structuring it at all. Furthermore, the model is illustrated with a flowchart diagram in which several entities are present, but no details are given on whether they are phases to carry out, actions or a zone delimitation.

A different approach is followed in [33], in which a very detailed six-phased methodology centered on privacy aspects of investigations is proposed that complies with the requirements of ISO/IEC 29100:2011 and follows the Enhanced Systematic Digital Forensic Investigation Model (ESDFIM). It covers the whole investigation process and does so with a reasonable degree of detail, complementing some of the phases with workflow diagrams. In this case, the whole concept depends on the installation of a piece of software named ProFiT, which is in charge of collecting and storing the information. However, not much information is provided on how an investigator should act in each of the phases.

A solution specifically designed for vehicles is presented in [26], which introduces a very detailed framework for the Internet of Vehicles (IoV). It focuses on providing guidelines for acquiring data, as well as storing it securely by using a distributed infrastructure. For this purpose, it is divided into two services: the "Forensics Gateway", which is a service embedded in the IoT device in charge of collecting the data, and the "IoV-Forensic Service", which stores the acquired data. In addition, it proposes an algorithm for verifying the integrity of the evidence collected, which is tested together with the framework in a simulated hypothetical scenario to evaluate the efficiency of the proposal.

The first approach to using new paradigms in IoT forensics is proposed in [7], which presents an investigation framework based on the principles of the Digital Forensic Research Workshop (DFRWS) [19] combining it with fog computing. It consists of six modules that are focused on detecting possible suspicious activity and, if this occurs, collecting the pertinent evidence. For these purposes, the authors develop a fog node that is connected to an IoT device, and the former filters and analyzes the data generated by the latter. Furthermore, the fog node notifies the rest of the devices in the network when a potential threat is detected, and stores the data from the affected nodes. To test the proposal, they present two theoretical use cases involving a smart refrigerator and a smart city. The work only addresses incident detection and, regarding the forensic process, the identification and acquisition phases. However, the authors mention that it would be ideal for the framework to be implemented as a middleware architecture, and used jointly with a methodology.

In summary, there are some interesting points that can be extracted regarding the use of external solutions in forensic processes, some of them being the following:

- They can be in the form of a server, a distributed infrastructure, a piece of software, or a single device.
- They can assist in tasks that demand a higher computational power than the one that IoT devices have. Evidence storage is one of the recurrent tasks for which these type of solutions are used.
- Their integration into the forensic investigative process can be done without altering it significantly, therefore still complying with the standards set by the forensic community and being suitable to be used in investigations that are part of a legal process.
- They are capable of performing proactive tasks that are interesting to decide when a forensic should start, such as incident and threat detection.

2.2 The Edge Computing Paradigm

When studying how the edge computing improves the performance of IoT networks [50] provides several arguments on its upgrades in terms of transmission, by offloading the data computation and storage of IoT devices, and monitoring and controlling traffic flow, storage, by balancing storing demands and leveraging load, and computation, by carrying out the complex tasks that IoT devices are not able to.

Detailing with more depth the edge computing paradigm, [5] explains its three typical technologies, namely mobile edge computing (MEC), cloudlets, and fog computing, and their application in the IoT. The first one focuses on working at the edge of cellular networks to gain computational capabilities. The second one provides computing resources to the network with low latency, and the latter allows to jointly work with the cloud to manage resources across networks.

While both edge computing and fog computing have the reduction of latency as a goal, they operate differently. The former uses computational resources at the edge of the network, while the latter introduces a layer between the network and the cloud, acting as an intermediary between them. As a result, edge devices interact with the traffic that is being generated in the network by directly connecting to the devices present

in it, working as close as possible to the data source. On the other hand, fog computing works together with edge devices to provide them with an infrastructure that connects them to the cloud.

The authors in [38] detail how the technologies above-mentioned have an impact in a specific context of the IoT: the industrial one. Some benefits of using these technologies are better resource allocation and fault tolerance, and large-scale real-time data processing. In addition, some application scenarios are mentioned, these being health management, intelligent connected vehicles, smart grid or smart logistics.

Focusing on its impact on cybersecurity, [35] detail the challenges and opportunities that come with the integration of edge computing in the IoT. Some of the challenges mention the high number of IoT devices and their lack of proper security measures, as well as highlight the concern regarding the privacy of the large amount of data that they exchange. When focusing on the opportunities, aspects such as using edge computing to execute machine learning algorithms for threat identification, and the possibility of enabling the use of lightweight encryption solutions to protect the data are pointed out.

When put into practice, the use of edge computing in cybersecurity fields has produced interesting results, especially when working in monitoring tasks for detecting attacks, anomalies, or intrusions. Proof of that is [18], which introduces a device called Shadow Security Unit, which is able to intercept the communications of Programmable Logic Controllers (PLCs) and Remote Terminal Units (RTUs) in order to analyze them.

Among the multiple functions which can be performed using this device, the generation of events, decoding of the protocol stream, capturing the state of the I/O probe modules and performing periodically check of the component operations are the most relevant ones.

However, other aspects such as data security can take advantage of the edge computing capabilities, as [24] shows, introducing a lightweight solution for encrypting the network traffic that is based on the Elliptic Curve Cryptographic (ECC), and that can be used on resource constrained devices, as it obtains similar encryption and decryption times to the conventional algorithm.

Similarly, [25] presents how edge frameworks can take advantage of the use of the Blockchain to provide a secure way of storing data, and protect them against alterations and modifications, addressing one of the main issues in the IoT: data privacy.

Although not centered on addressing IoT forensics, but focused on the Industry 4.0, [40] presents a framework using intelligent edge computing for the detection of attacks. In order to do this, a library with attack patterns is used, and, when an attack is detected, a module is triggered that safely stores the data that raised the alarm, which is encrypted and stored in a database. In addition, when compared to other frameworks, it can be seen that this proposal obtains better results. However, in terms of the forensic relevance of the data that is gathered, almost no detail is provided on them, and the framework is not tested in any forensic scenario, so there is no information on how it would perform in an examination.

Ultimately, the edge computing paradigm has emerged as an interesting concept that may be capable of assisting in the forensic process. Some of the key points that can be extracted after studying the related work that focus on it and its impact in the IoT are the following:

- It provides a more powerful environment in which to perform certain tasks such as secure storage, encryption, redundancy, or processing large-scale data, prompting new features that could be included in forensic investigations.
- One clear example in which the use of the edge computing has proven to be quite useful is for detecting threats and attacks in a network, which can lead to the opening of a forensic process.
- Since an edge device is part of the IoT network, it is capable of having access to the data generated in it, and, consequently, the sources of evidence in an investigation, meaning that the way in which the investigator can interact with the pieces of evidence changes.

Finally, a comparison between this proposal and the ones presented by the research community is presented. In this case, methodologies, models, frameworks, or guidelines that are centered on addressing the IoT environment and use any kind of service, piece of software or hardware to carry out or assist in the investigation process are reviewed. In Tables 1, and 2 a summary of the comparison is presented, although the main aspects are described below:

- Our proposal is not centered in a specific IoT context, on the contrary, it can be used in any of them. Its limitation is based upon the compatibility of the edge node with the protocols being used in the network, but not by the context in which the IoT network is being used.
- It is based on the edge computing paradigm, which has shared characteristics with the IoT, one of these being the devices that are used in them, an aspect that facilitates the interoperability and integration of the IoT and the edge, thus naturally addressing

Table 1: Summary of the comparison of the proposal with previously existing ones (I).

Proposal	Approach	Process	Monitoring	Identification	Acquisition	Analysis	Limitations
[36]	Hadoop server	Reactive	✗	By studying the machine to machine communication	Live data extraction	Conventional approach	The Hadoop server only stores the data gathered by the investigator
[18]	Physical unit	Proactive	Monitors the interaction between devices and their I/O modules	✗	✗	✗	Needs to be attached to the devices
[33]	Piece of software installed in the device	Reactive	✗	✗	Logical acquisition of the data generated by the software	Data collected by the software	Every device to be studied needs to have the piece of software installed beforehand
[26]	Distributed central platform and a service installed in each device	Reactive	✗	✗	Logical acquisition of the data generated by the software	Data collected by the software	Focuses on the interactions of the vehicle with other entities, but not in the data present in the vehicle
[7]	Fog node	Reactive and Proactive	By detecting patterns in the communications	✗	Records all traffic data exchanged	✗	Only focused on the communications between IoT devices
Proposed research	Edge node and optional cloud access	Proactive and Reactive	Anomaly detection by studying the network traffic and each device storage	Based on the data collected by the node	Physical or logical performed by the edge node in addition to network traffic	Data collected by the node	The compatibility of acquiring the storage from every IoT device is not guaranteed

some of the challenges found in the former, such as the lack of processing and storage capabilities. This makes it a more feasible approach than using traditional solutions like distributed platforms, smartphone, or pieces of software.

– It covers the whole investigation process and does so from a practical perspective, offering detailed guidelines on how to proceed during each phase.
– The forensic solution that is used to shape this methodology is not a piece of software, server, or platform but an IoT device, which means that it is more likely for this device to be compatible with the protocols being used by IoT devices, as well as makes it easier its handling and inclusion into the network. In addition, since the edge node has access to the IoT network, it does not rely on an external connection in order to work, only to connect with the cloud, which may not be always present in the topology.
– Our proposal addresses both the proactive and reactive aspects of IoT forensics, thus allowing the monitoring and detection of anomalies in an IoT network and also initiating the investigation process once that an anomaly has been detected.

Table 2: Summary of the comparison of the proposal with previously existing ones (II).

Proposal	Context	Practicality	Versatility	Types of Evidence Gathered	Evaluation
[36]	Any	Low	Medium	Network traffic	✗
[18]	Industrial Control Systems	Medium	Low	Network traffic and non-volatile memory	✗
[33]	Any	Low	Low	Non-volatile memory	Hypothetical case study
[26]	IoV	Medium	Low	Network traffic	Hypothetical case study
[7]	Any	High	High	Network traffic	Theoretical case study
Proposed research	Any	High	High	Non-volatile memory, volatile memory, and network traffic	Comparison with existing models and practical evaluation

– By interacting with the devices in the network and analyzing the traffic exchanged in it, the process of tracking the possible sources of evidence in an investigation becomes quite straightforward.
– It allows working with the three main types of forensic evidence, namely non-volatile memory, volatile memory, and network traffic, while the related work only focuses on one of them or two at most.
– The data captured by the edge node are the ones generated by the IoT devices present in the network, meaning that the investigator can have access to the original raw data, and they are not accessing the logs generated by an application. In addition, this methodology covers both the collection of the network traffic, as well as the non-volatile memory of the IoT devices that allows so, and does this by following a remote acquisition method.
– Using these data, the investigator has access to enough information to be able to carry the analysis process. Furthermore, alternative options are provided in case the acquisition fails for any of the sources of evidence. For example, this means considering the remote analysis as a viable process, for which the edge node can be used.
– It introduces a new phase destined to address the process of bringing back the IoT network to a functioning state, which is a task that is usually requested in private forensic investigations and which can be done easily thanks to keeping a history of the changes in the storage.
– Unlike the related work, the proposed methodology is compared with the solutions presented by the research community and evaluated from a practical standpoint in two real-life situation scenarios, this serving as a way of determining its feasibility, effectiveness, versatility, and completeness.

2.3 Starting Point

In order to ensure the effectiveness of the proposal, this research uses as a reference an IoT forensic methodology that has already been accepted by the research community. This proposal is [16] which presents a concept IoT forensic methodology, with its main aspects being the following:

– It uses a traditional forensic methodology as a reference, namely [51], in order to ensure the effectiveness and feasibility of the proposal. It gathers the common processes shared by all the forensic models proposed since 1984 and generates a generic one. At the same time, with the goal of adapting to the requirements of IoT investigations, it also extracts aspects from [15], which presents a context-centered IoT methodology from which general ideas can be extracted.
– It is divided into the following phases: "Pre-Process", "Identification", "Acquisition & Preservation", "Analysis", "Evaluation", and "Presentation and Post-process".
– The identification phase is carried out by studying the logical communications made by the IoT devices. Once that the sources of evidence have been determined, an order of relevance is established for all of them.
– The acquisition phase is centered on acquiring the non-volatile memory, volatile one, and network data. In addition, it establishes an order for performing the different techniques depending on the impact that they have in the integrity of the data.
– The analysis phase addresses the possibility of carrying out a remote examination either when it is not necessary to maintain the integrity of the evidence or when the acquisition is not feasible.
– A new phase is added, namely "Evaluation", which aims to study the pieces of evidence and the conclusions drawn from them during the analysis phase,

but from the viewpoint of the whole IoT environment, and not from an individual one.

Even though there is a fundamental difference in terms of how both methodologies are modelled, in order to demonstrate that this proposal is not just an extension of the starting point, the authors' believe that it is crucial to emphasize the differences between them so as not to lead to confusion. With the aim of summarizing the information detailed below, Table 3 is presented.

- The most immediate difference, as mentioned, is the core concept upon which the methodology is modelled. The reference proposal relies on conventional techniques and tools that are adapted, to the extent possible, to the requirements of IoT investigations. Therefore, although brief guidelines are provided on how to approach the examination process, the execution of the phases depends on the investigator's ability to do so. However, in this proposal, the inclusion of the edge node aims to reduce the dependability on the investigator by providing a tool to assist, automatize, and facilitate some of the most crucial phases of the investigation. This results both in a reformulation of these phases, as well as in an extension of the details provided in the reference proposal.
- The proposed methodology adds the proactive approach as an additional way of tacking the forensic investigation. This means changing the procedure to follow in many phases in order to adapt it to the fact that the amount of data for the investigator to study in a proactive scenario is quite different from a reactive one. In addition, two additional phases, namely "Detection" and "Recovery", are created to model this proactive approach.
- The "Identification" is carried out by studying the logical communications made by the IoT devices in both proposals, but in this one the task is performed by using the edge node. Furthermore, if used in a proactive approach, the investigator also have access to the history of the behaviour of the network.
- Although the acquisition techniques mentioned in both proposals are the same, and so are the types of evidence that are handled, the reference methodology does not cover in detail the practical aspects of this phase. In addition, with the inclusion of the edge node, the way of approaching the remote acquisition of the non-volatile memory, the volatile one, and the network traffic varies, which produces substantial changes in the way in which this phase should be tackled.
- Finally, the reference proposal is not assessed in any way, while the authors' methodology is compared with the related work, and evaluated from a practical viewpoint by applying it in two IoT forensic scenarios.

3 Methodology Description

The use of edge computing in this methodology means including a new node in the IoT network which is capable of interacting with the devices that are present in it, as well as with ones outside it. This allows it to access all data that is being exchanged at any given moment in the IoT network and also the information stored in the devices. In addition, the possibility of exchanging information with systems outside the IoT network means that these data can be sent to third parties in order to monitor their state, and that other services can be used to complement the lack of computational power that this edge node has.

In terms of which device would be suitable to perform these operations, a board such as a Raspberry Pi [39] could be a good example. It can perform reasonably demanding tasks and can deploy several services for third parties to connect to it, as it can execute complex operating systems, such as Ubuntu Core [13]. In addition, it is crucial to ensure that the device that will act as an edge node is compatible with the protocols that are being used in the IoT network, so that the communications between devices can be analyzed. Both the operating system and the board mentioned complies with this requirement.

Another key element that can be part of this methodology is the cloud, which can compensate for the lack of computational power of the edge node. For example, it can store the captured data and execute the incident detection algorithm, aspects that are detailed in this section. A graphical example of the resulting environment can be seen in Figure 1.

With respect to the phases that comprise this methodology, two new ones are included. The first one is "Detection", which is added due to the possibility of performing preventive actions by using edge computing to monitor the state of the IoT network. Secondly, the "Recovery" phase is introduced to cover the process of bringing back the IoT network to a functioning state using the data collected by the node. On the other hand, it must be noted that the "Evaluation" and "Presentation & Post-process" phases do not significantly change with respect to the reference methodology, as the edge node cannot assist in the processes carried out in them. However, with the aim of giving the highest degree of detail possible and readability, they are detailed in this article.

Proposal	Approach	Process	Evaluation	Feasibility	Level of Detail	Reference	Limitations
[16]	Adaptation of conventional procedures to the IoT	Reactive	✗	High	Low	[51,15]	Follows the traditional investigator-oriented concept
Proposed research	Integration of an edge node to assist on IoT investigations	Proactive and reactive	Theoretical and practical	High	High	[16]	Limited by the edge node's functionality

(a) General overview comparison.

Proposal	Detection	Preprocess	Identification	Acquisition & Preservation
[16]	Not considered	Design the action plan using static information	Based on logical communications	Considers all three types of evidence using conventional techniques. The preservation is done following a traditional approach using hash codes
Proposed research	Network traffic and file system monitoring	Design the action plan using static information and dynamic one gathered by the node	Based on the information extracted by the node after studying the logical communications	Considers all three types of evidence and uses the edge node as a tool to perform the evidence collection. The preservation is done by automatically generating logs with the checksum information and acquisition data

(b) Phase to phase comparison (I).

Proposal	Analysis	Evaluation	Presentation & Post-process	Recovery
[16]	Offers guidelines on which technique to perform and their advantages and disadvantages	Draws conclusions from the perspective of the whole environment	Covers the closing aspects of the investigation	Not considered as an independent phase. Offers brief guidelines in the previous phase for cleaning and restoring the environment
Proposed research	The edge node offers information on the feasibility of performing a remote analysis	Draws conclusions from the perspective of the whole environment	Covers the closing aspects of the investigation	Uses the data gathered by the edge node to restore the environment and check whether it is functioning properly again

(c) Phase to phase comparison (II).

Table 3: Comparison of the proposal with the starting point.

As a result, the methodology is divided into the following phases, which are individually described below:

- Detection: the preventive tasks of monitoring and incident detection are carried out.
- Pre-process: the investigator designs the actuation plan and performs the necessary task prior to the beginning of the investigation.
- Identification: the investigator determines which elements present in the scene are likely to contain evidence and how to proceed to their acquisition and analysis.
- Acquisition & Preservation: process involving capturing and preserving the data from the sources of evidence.

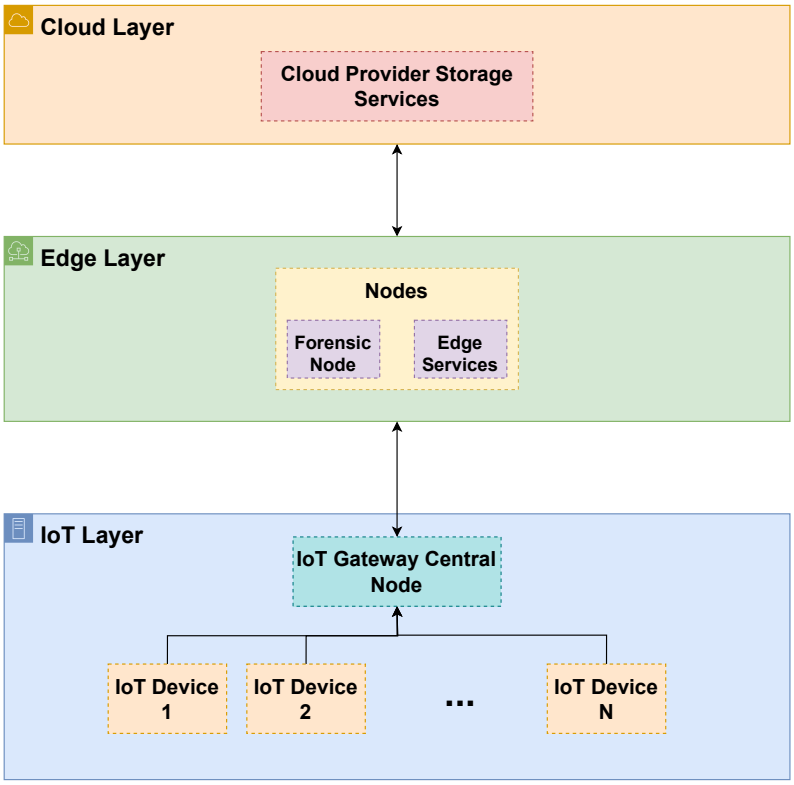

Figure 1: Graphical representation of the environment.

- Analysis: the investigator draws individual conclusions from the data from the sources of evidence.
- Evaluation: all the pieces of evidence extracted are reviewed and conclusions are drawn from the perspective of the environment.
- Presentation & Post-process: the results from the analysis are presented, and the actions needed before closing the investigation are carried out.
- Recovery: the IoT network is brought back to a functioning state in case that is needed.

Operating Modes. A novel aspect of this proposal is that the methodology can be used in two different approaches, which are detailed below.

- Proactive: considers a scenario in which the proposal is followed before the investigation process begins, with the edge node operating both as a monitoring tool and a forensic one, and making the IoT network a forensic-friendly environment, as well as offering a higher degree of forensic-readiness. This way, the methodology would start with the "Detection" phase, and the edge node would be able to detect any anomalies in the network traffic or in the devices' file system, with the possibility of automatically starting the forensic investigation when this happens, reducing the response time to a minimum.
- Reactive: the methodology is used as any other common forensic model when an investigator is asked to perform an investigation. Therefore, the application of the methodology would start with the "Preprocess" phase.

Under these circumstances, the proposal can be used in the two main scenarios in which normally a forensic investigation is required. The first and most common one, when an investigator arrives at the scene with the aim of starting an investigation, independently of the initiator of the process being the police or a private investigator. The second one, in the context of incident response, with this methodology providing a tool that automatically starts the forensic process.

3.1 Detection

In this phase, the network communications and the state of the devices are checked in order to determine whether they are behaving correctly.

With the edge node being compatible with the most commons IoT protocols, the data that are exchanged in the IoT network is studied and collected in real-time. This operation can be easily performed and is transparent to the network's performance, and allows tracking

aspects such as the devices in the network, the protocols being used, the packets exchanged in it, and the connections made by each device.

In addition, if the device allows it, the edge node can connect to it using a remote service such as SSH or Telnet, which can allow the monitoring of the data that is stored in it. However, this has a great impact on the device's performance, as executing acquisition tools like dd [17] is a demanding task, so these operations should be scheduled with this in mind.

All this information is studied by the edge node, but also stored in it. Consequently, the investigator has access to the raw data acquired as well as the features extracted by the node, which are listed in the "Identification" phase.

As the node has access to these data, it is possible to carry out monitoring tasks with the aim of detecting an incident in the network. In this proposal, we suggest using anomaly detection techniques, which, as shown in [12], [6], [45] or [31], can lead to successfully detecting an abnormal device behaviour. This way, if an anomaly is detected, the edge node notifies the network manager and starts performing the tasks needed to ensure that the possible pieces of evidence are not lost, an operation that is described below. The monitoring task could also be carried out by the cloud if the edge node is not able to manage such a great amount of data. In this case, the edge node would forward the data to the cloud and then would receive the result from their analysis.

As a result, the possibility of having access to a device that is constantly monitoring the data that are exchanged in the IoT network means that the response time when an incident arises can be reduced to a minimum. Furthermore, the raw monitored data can be used as a source of evidence to perform the examination process.

3.2 Pre-process

In order for the investigator to prepare their action plan, it is crucial to determine, among other aspects, the number of devices in the IoT network and their technical specifications so that they can decide how to approach their examination.

This task can be automatically performed by the edge node, which can study the network data and extract this information. In addition, it can also check the remote accessibility of the device by attempting to connect to it, which is extremely useful for the investigator in order to decide which methods would be interesting to execute during the acquisition and analysis phases. All these data are captured, stored and analyzed, presenting it to the investigator.

In addition, the fact that the edge node can connect to the devices in the IoT network offers the possibility for the investigator to execute an action over them if needed. For example, if a malware sample is suspected to be involved in the incident, the investigator could send the order to shut down the devices, and this could stop the malicious program from spreading over the network.

Reactive mode. If the methodology is being followed in its reactive mode, this is the phase in which the edge node is set up in order to start interacting with the devices in the network and their data.

Proactive mode. If the methodology is being followed in its proactive mode, it should be noted that the investigator also has access to the data gathered during the "Detection" phase, so they can easily check the network's behaviour history.

Other relevant tasks that should be performed in this phase are: to learn the nature of the incident, to determine the level of forensic soundness required in the investigation, and to request any data that the cloud provider may be storing if the IoT network is using that type of service.

3.3 Identification

The task of determining the range of the investigation and identifying the sources of evidence is simplified thanks to the edge node. As mentioned above, it is capable of tracking the devices in the network, as well as the communications that are made between them. In particular, the following features are extracted:

- The protocols being used in the IoT network and the number of packets exchanged in each of them.
- The devices in the network, their ID, and, when available, their model.
- The number of packets sent and received by each device.
- The number of connections made by each pair of devices.
- The date and time of the last packet sent by each device.
- Whether a device can be remotely accessed and, if so, the file systems that are present in its non-volatile memory.
- If used in the proactive approach, the connections and packets that have been detected as an anomaly.

Consequently, it is extremely simple to know the number of devices that are part of it, whether they are still active, or whether their remote acquisition is feasible. On top of that, any interactions that the investigator may need to perform can be done through the

edge node, and with the data analyzed and stored by it, instead than with the IoT devices in the scene, thus reducing the chances of compromising the integrity of the sources of evidence.

Furthermore, having access to the communications made in the network also allows the investigator to determine the relevance of the different devices in it, therefore facilitating the process of prioritizing one device over others.

Under these circumstances, this phase consists on studying the information that has been gathered during the "Pre-process" phase in order to determine which devices in the scene might contain relevant data for the investigation. In addition, thanks to knowing how feasible the acquisition of a device is based on whether it is remotely accessible, the investigator has enough data available to decide how to approach the acquisition and analysis of each device individually.

3.4 Acquisition & Preservation

The investigator gains a higher level of flexibility when facing this phase, as they can combine traditional methods with the features provided by the edge node. The sources of evidence that can be acquired are the following:

- Non-volatile memory. Collecting this type of data is more difficult than in conventional scenarios due to is being soldered to the IoT device's board. However, there are several techniques that can be performed, as it is described below.
  - Physical acquisition methods. Conventional acquisition methods, such as extraction and acquisition (if the storage is in the form of a microSD card or a drive), JTAG/UART or ISP, or chip-off, have been confirmed to be effective approaches to collect the non-volatile memory, as seen in, [29], [11], [48] and [20].
  - Remote acquisition using the edge node. The first relevant information provided by the node is the feasibility of performing this technique. If the device allows it, the node can connect to it using SSH or Telnet, list the file systems available, and perform the acquisition. By default, it is stored in the node, but it could be sent to the cloud as well. Given the vulnerability of the Telnet protocol against eavesdropping attacks such as Man-in-the-Middle (MitM), SSH should be prioritized over Telnet if both services are available. This way, confidentiality is guaranteed by using a secure encrypted channel.
- Volatile memory. Unfortunately, performing a remote acquisition of the volatile memory is an almost impossible task, as, at the moment of the design of this proposal, there are no compatible IoT tools which allow so. Consequently, the remaining options are carrying out physical techniques such as debugging or JTAG/UART, but they require having physical access to the device, and will only provide access to the raw contents of the memory, which is extremely complex to analyze.
- Network traffic. In this case, the most logical procedure to follow is using the edge node to perform the acquisition, as it is already configured to do so. Otherwise, the investigator would need to set up its own tools and make sure that they are compatible with the protocol that is going to be collected. If none are available, they should opt for executing the acquisition in a router or the IoT gateway, but this requires for them to be remotely accessible.

By including the edge node in the IoT network a successful acquisition process is guaranteed, something that is not always assured when carrying out IoT forensic investigations, meaning that, at least, the network traffic will be collected, thus having some data from which to draw conclusions later. After that, they can decide whether it is necessary to perform a physical acquisition, either because there is no other option, or to complement the data obtained by the edge node.

Proactive mode. In the proactive operating mode, once that an anomaly has been detected in the system, the edge node begins to automatically carry out the acquisition process on the devices to which it has remote access to. These data is paired with the network traffic to study in the analysis phase.

Reactive mode. When used in the reactive mode, the same actions are carried out, but the process is manually initiated by the investigator, since there is no trigger that automatically activates the acquisition phase.

Data preservation. Independently of the type of data acquired, they are safely stored, and a log file is generated with the date and time in which the acquisition was made, the hash code associated with it, and the name and model of the captured device.

3.5 Analysis

In this phase, the goal is to individually study each source of evidence in order to draw conclusions that can lead to determine what occurred in the incident. There are two techniques that can be executed to examine a

device. The first one, and traditionally the most common one, is to carry out an offline examination studying the evidence collected in the previous phases. The alternative option is performing a remote analysis, which requires directly executing the corresponding tools in the device to be examined. Opting from one or the other depends on the result of the acquisition phase, as well as in the need of preserving the integrity of the evidence. Taking into account the current standards when it comes to evidence preservation, an offline approach should be prioritized over a remote analysis. However, given the difficulty of successfully performing a physical acquisition on an IoT device, the latter is expected to become a more recurrent approach than in conventional scenarios.

Even though the analysis process mainly depends on the ability of the investigator and the possibility of executing the necessary tools to extract knowledge from the sources of evidence, the edge node can assist in the following tasks:

- Determine the feasibility of performing a remote analysis. If the edge node has not been able to extract data from a device, this means that it is not possible to interact with it, so the investigator will not be able to perform a remote analysis.
- Examine the data gathered in previous phases. If data were captured from a specific device, they can be used to draw conclusions. In the worst case scenario, the investigator would have access to the network traffic.
- Provide access to a history of data. If used in the proactive approach, the node would have stored data on how the network was behaving before the incident arose and, in some cases, specific non-volatile and volatile device data.

Therefore, even though the edge computing node does not facilitate the process of drawing conclusions as such, it can help the investigator by providing them with more information that would help them decide on whether to perform a remote analysis, an offline one or whether it may be necessary to rely on other devices to extract information of the device that is being analyzed.

3.6 Evaluation

The high interconnectivity of IoT networks increases the likelihood of incidents affecting several devices. Consequently, a new phase is necessary to confirm all collected evidence, determine links between them, and interpret the results from a holistic perspective to establish the incident's cause with supporting evidence.

In order to do so, each piece of evidence gathered during the analysis phase is assessed to determine its impact on the system and whether it affected other devices. This not only may lead to drawing better conclusions on what occurred during the incident, but also may allow finding new evidence or piece existing ones together that did not make sense when studied individually.

Once that all pieces of evidence have been evaluated, the final task is to study the linked ones from a holistic perspective, instead of from a device-centered one, with the aim of drawing conclusions regarding the incident's cause. The outcome of this phase should be the ability of the investigator to chronologically retrace the incident's actions, supported by concrete evidence, and determine the extent of the impact on the network's devices.

3.7 Presentation & Post-process

This phase consists on presenting the conclusions drawn from the examination in a clear and understandable manner. In order to do so, the following processes are carried out:

- Writing and presenting the forensic report: typically, a detailed forensic report is the result in which the findings are presented to the client, whether it is a private one or a court of law. It outlines the investigation's objectives, methodology, findings, and conclusions providing supporting evidence. In addition, it may be necessary to explain the findings of the investigation and answer any questions, either in court or in a private meeting with the client.
- Returning the original sources of evidence: involves the process of restoring any physical or digital evidence that was collected during the investigation to its rightful owner or custodian. This process is critical to ensure the integrity of the evidence and to comply with legal and ethical obligations. In some cases, it may be necessary to destroy the sources of evidence instead of returning them.

3.8 Recovery

The final phase of the methodology aims to bring back the IoT environment to a functional state, a task in which the edge node plays a critical role.

Firstly, since it is able to study the network traffic, it can evaluate whether the cause of the incident is still present by detecting any anomalies in it. Secondly, when used in the proactive mode, the task of restoring

the system to a previous state can be fairly simple and quick if the node was able to access it remotely during the "Detection" phase, as complete images of the file systems would have been acquired. If this was not possible, or the methodology is being followed in the reactive approach, the restoring data should be provided by a system administrator or by the vendor, if available. Once that this task is completed, the monitoring process will automatically evaluate the effectiveness of the actions performed, and will notify the manager if it detects any residual anomaly. In addition, since in the proactive mode the investigator can access the communications when the network was behaving correctly, they can compare both situations to determine whether the environment is functioning properly again.

Overall, these actions are necessary to ensure that all traces of the incident are removed, the affected systems are restored, and measures are put in place to prevent similar incidents from occurring in the future.

As a summary of all the most relevant operations that are carried out in the phases that comprise the methodology, a graphical representation of the workflow is presented in Figure 2.

4 Practical Evaluation of the Proposed Methodology

To evaluate the proposal in a practical scenario, two case studies are presented in which the developed methodology is used in a forensic investigation process.

The goal of these case studies is to show the feasibility of the proposed methodology, proving that the processes mentioned on it can actually be performed. In order to do so, a Raspberry Pi Model 3+ [39] is used that runs the Ubuntu Core [13] IoT operating system. In it, all the features described above are available for the investigator to execute by remotely connecting to it through SSH. In addition, for the first case study, a Tmote sky module [32] and a CC2531 stick [44] that connect to the Raspberry via USB are used to work with the Zigbee traffic, as they are compatible with the IEEE 802.15.4 standard. To acquire and analyze the traffic, Zigbee Tools [1] and the Scapy framework [4] are used, while the Paramiko library [23] is used to remotely connect to the IoT devices.

The two scenarios presented are designed taking into account that the smart home and the IIoT are two of the most common contexts in which to find IoT devices [42,2,3]. Furthermore, they depict environments in which different types of IoT devices and systems can be found, which allows us to test the versatility of the proposal and how it performs under different challenges.

4.1 Smart Home Investigation

This first case study aims to evaluate the behaviour of the methodology when used in a reactive context. Therefore, the investigation takes place once that the client has requested the investigator to perform a forensic analysis, meaning that the phase which has an eminent proactive approach, namely "Detection", is not covered in this case.

Regarding the scenario in which the investigation is taking place, the IoT system present is a "Xiaomi Mi Smart Sensor Set" [49], destined to be used in a smart home context. Its topology can be seen in Figure 3.

Pre-Process. Due to the reactive aspect of the case, the investigator only has the information that has been provided by the client beforehand, so their assumptions must be made from a theoretical standpoint by analyzing the technical specifications of the kit using provided by the manufacturer. By doing so, the following conclusions can be drawn:

- The kit is comprised by six devices:
  - The Mi Control Hub, which acts as a central node.
  - Two Mi motion sensors.
  - Two Mi windows and door sensors.
  - A Mi wireless switch.
- The user connects to the control hub by using a mobile app, namely "Mi Home". This connection is made through Wi-Fi.
- The data exchanged between the control hub and the sensors is sent using Zigbee.
- None of the devices in the smart home kit have a removable storage. All of them have a soldered one. In the case of the "Mi Control Hub", there is a dedicated memory chip for the storage, but the sensors use the memory of their Zigbee module to perform their tasks, so there is not a storage as such. This means that very little information can be stored in the sensors, and that the only feasible acquisition method would be a remote one, since the physical techniques are not compatible with a Zigbee module.
- There is a possibility that the control hub may be sending data regarding the kit to the Xiaomi cloud, so it may be necessary to make a request to the provider.

However, the investigator does not have access to the dynamic information of the IoT network, therefore they cannot know the actual number of devices in it at any given moment, their model or whether they are powered on. Upon arrival at the scene, the edge node is inserted into the IoT network.

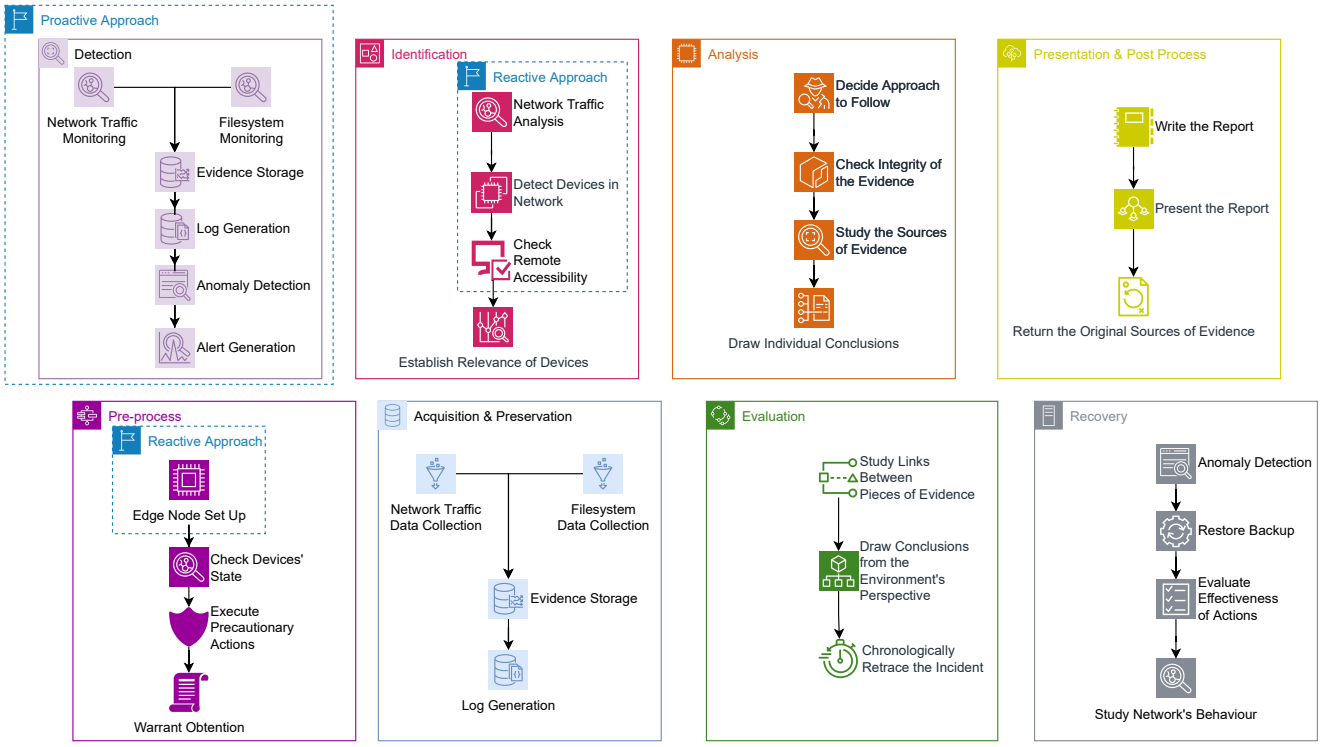

Figure 2: Summary of the main actions carried out in the methodology.

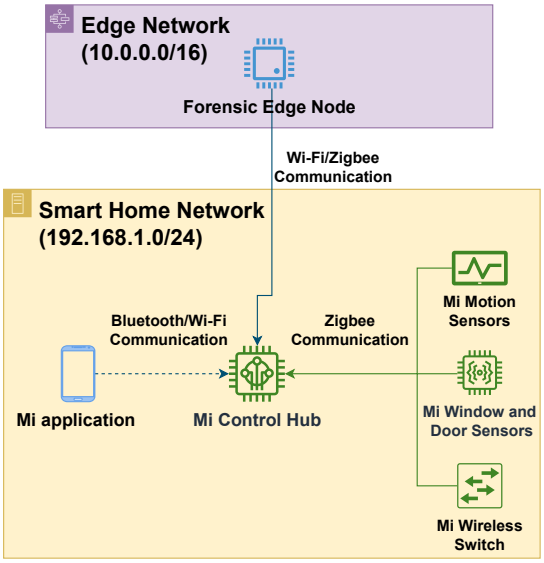

Figure 3: First case study's topology.

Identification. This is the first phase in which the information collected by the edge node makes its appearance in the investigation process. Once that the node is ready, it starts acquiring the network traffic that is being exchanged in it. In this case, the protocol used to exchange data between the devices of the kit is Zigbee. After interpreting the data collected, the investigator gets information regarding the devices which have made any communication since the edge node was included in the IoT network, as well as how many packets have been sent and received by each one of them, and when was the last time that a device establish a communication, as can be seen in Figure 4.

Upon inspection of the information provided by the edge node, it can be seen that the number of devices identified when analyzing the network traffic matches the number of devices which comprise the smart home kit. Consequently, it can be assumed by the investigator that all the devices of the kit exchanged information when the capturing process started, and that there were not any other devices not belonging to the aforementioned kit that generated traffic.

In addition, information with respect to the behaviour of the network can be extracted. Given the number of packets received, it can be concluded that the Mi Control Hub is the device with ID "0x0000", that all its sent packets are broadcast, and that it is the device which receives the packet sent by the rest of the devices, namely the sensors. Unfortunately, there is no other way of automatically determining which Zigbee ID is associated with each IoT device, as there is no field in the Zigbee protocol allowing a clear univocal identification, but by analyzing the captured traffic, it

```
The following devices have been detected:

  Device ID
0    0x0000
1    0xf969
3    0xf0e2
4    0x7d8e
5    0x14a4
6    0xae98

The number of packets sent by each device is:

        Packets Sent
Source
0x0000            38
0x14a4             1
0x7d8e             1
0xae98             2
0xf0e2             6
0xf969             8

The number of packets received by each device is:

             Packets Received
Destination
0x0000                     18
Broadcast                  38


The last packet sent by the device with ID 0x0000 was in 2022-11-21 20:09:13.274963 UTC Time
The last packet sent by the device with ID 0xf969 was in 2022-11-21 20:03:48.753545 UTC Time
The last packet sent by the device with ID 0xf0e2 was in 2022-11-21 19:46:57.397198 UTC Time
The last packet sent by the device with ID 0x7d8e was in 2022-11-21 19:18:01.461335 UTC Time
The last packet sent by the device with ID 0x14a4 was in 2022-11-21 19:22:47.408199 UTC Time
The last packet sent by the device with ID 0xae98 was in 2022-11-21 19:59:03.604706 UTC Time


Acquisition started: 2022-11-21 19:05:51.192045
Acquisition ended: 2022-11-21 20:10:28.191510
File generated: Capture_2022-11-21_190551.192045.pcap
MD5 cheksum: 378e2df429a3ce6b0bd1414e566efedd
SHA1 checksum: 47cddb84a505cc78ece50a27f2449a550e848bea
```

Figure 4: Information provided by the edge node after capturing the Zigbee network traffic.

is possible to match each ID with each device, obtaining the result shown in Table 4.

Table 4: Information extracted after the analysis of the Zigbee traffic.

Device	ID
Mi Control Hub	0x0000
Mi Window and Door Sensors	0x0f0e2 and 0xf969
Mi Motion Sensors	0x07d8e and 0xae98
Mi Wireless Switch	0x14a4

In the same way, the edge node is able to capture the Wi-Fi traffic exchanged in the IoT network as well. In this case, as mentioned above, the only IoT device capable of connecting through Wi-Fi is the central node, so the number of devices detected is lower than when working with the Zigbee traffic. In this case, as can be seen in Figure 5, there are three devices: the edge node, the smartphone managing the IoT set, and the router. Through the study of this protocol, the investigator is able to confirm that there is an active connection with an external address, which belongs to the Xiaomi cloud, thus adding another item to be considered as a source of evidence in the investigation. In addition, it also ratifies that there is a mobile phone being used to control the smart home kit, so it must also be studied as well.

Acquisition & Preservation. In order for the edge node to carry out the acquisition, a connection needs to be established between it and the desired IoT device to execute the proper commands. In this scenario, the only device that would allow a remote connection would be the "Mi Control Hub", as it is the only one using a Wi-Fi connection.; the rest of the devices only use the Zigbee protocol to send data. However, since the Mi Control Hub is not executing any of the services that would allow launching a remote terminal and performing the acquisition, this option must be discarded.

As a result, as none of the identified devices can be accessed through a remote connection, the only way to

```
The following devices have been detected:

         Device ID
0    192.168.137.1
1  192.168.137.249
2    18.159.88.239

The number of packets sent by each device is:

                Packets Sent
Source
18.159.88.239             13
192.168.137.1             42
192.168.137.249           17

The number of packets received by each device is:

       Destination  Packets Received
0    18.159.88.239                13
1    192.168.137.1                 4
2  192.168.137.249                17
                                          2022-11-21
The last packet sent by the device with IP 192.168.137.1 was in 2022-11-21 19:06:28.191510 UTC Time
The last packet sent by the device with IP 192.168.137.249 was in 2022-11-21 19:06:26.427034 UTC Time
The last packet sent by the device with IP 18.159.88.239 was in 2022-11-21 19:06:26.580564 UTC Time


Acquisition started: 2022-11-21 19:05:51.192045
Acquisition ended: 2022-11-21 19:06:28.191510
File generated: Capture_2022-11-21 19:05:51.192045.pcap
MD5 cheksum: 1c31935985c2907c13ceecd0414bc48b
SHA1 checksum: b99dc20f1343b5aa9e2f3f2f32c19c21e66d2979
```

Figure 5: Information provided by the edge node after capturing the Wi-Fi network traffic.

perform a storage acquisition is to opt for a chip-off or a JTAG. Therefore, the edge node cannot collect any data of the storage. On the other hand, as demonstrated in the "Identification" phase, it is able to sniff the network traffic generated in the IoT network. Due to the fact that this is a reactive process, the same traffic captured in said phase can be used as an evidence. Consequently, although the edge node allows so, it is not necessary to perform another network acquisition, as the state of the IoT environment is not likely to change.

Regarding the preservation of the network capture, the edge node stores the resulting file and, in order to guarantee the forensic soundness of the evidence, generates a log file in which information regarding its hash code, the name of the captured file and the date for when the acquisition started and finished. In Figure 6 an example of the mentioned log file is presented.

Analysis. Once that the pieces of evidence have been acquired, the rest of the investigation process relies on almost solely in the investigator's ability. However, as the edge node has not been able to remotely connect to any of the devices of the IoT network, the investigator has the information that a remote analysis cannot be carried out in any of them. In addition, since these devices do not allow direct interaction, the online method is incompatible as well. Although it is a simple piece of information, it means discarding these methods right away instead of having to evaluate each device individually, which would result in a longer process.

With respect to the data stored by the cloud and the mobile phone, since the edge node has no impact on how to conduct the analysis of these sources of evidence, and the goal of this section is to present what effect the edge node has in an investigation, only a brief description is provided. With respect to the cloud, there is no way to extract valuable information from the user's account, thus the investigator would need to contact the provider to ask for the data that it has stored, with the difficulties that this process entails. However, the smartphone does store useful data such as the logged actions by the sensors, connected devices or a cache of the landing page of the app.

Conclusions. In order to highlight the effects that the edge node has had in this case study, the main conclusions that can be drawn after this test are summarized below:

- The edge node has provided information regarding the activity of each device in the network for all the protocols being used in it, including that of when was the last time that each one of them generated traffic. This allows the investigator to know how many devices are in the network, their IDs, and helps to understand the behaviour of the environment. With respect to the first piece of information, it provides so without having to analyze the data stored by the smartphone app, which would require to root the device and either acquire its storage or perform an online analysis.

```
:/# cat Capture_2021-02-10_190213.062753.log
Acquisition started: 2022-11-21 19:02:13.062753
 Acquisition ended: 2022-11-11 20:09:13.274963
 File generated: Capture_2021-11-21_190213.062753.pcap
 MD5 cheksum: 378e2df429a3ce6b0bd1414e566efedd
 SHA1 checksum: 47cddb84a505cc78ece50a27f2449a550e848bea
```

Figure 6: Log generated by the edge node for the pcap file collected.

- The network traffic captured by the edge node has served as a piece of evidence that can be analyzed by the investigator. The file containing all the traffic is stored in the edge node, calculating its hash code and logging the beginning and ending dates of the capturing process, thus assuring its forensic soundness.
- The edge node fails to perform a storage acquisition as none of the devices of the network allow remotely connecting to them, so it would not be successful either if carried out manually by an investigator. This highlights the importance of at least having access to the network traffic, as the only methods that could be used to acquire the storage, namely JTAG and chip-off, do not always succeed and are quite complex. Therefore, if both methods fail, the investigator would not have any pieces of evidence to analyze.

4.2 IIoT Investigation

In this second case study, the goal is to represent the behaviour of the methodology when it is used in a proactive scenario. In addition, the context in which the investigation is taking place also changes in order to show the flexibility of the proposal now being an IoT system destined to be used in the IIoT. In particular, the IoT network is comprised by the following devices:

- A central node which deploys the services provided by the IoT network.
- Two sensors communicating between each other and the central node using the Modbus protocol.
- Two sensors communicating between each other and the central node using the Message Queuing Telemetry Transport (MQTT) protocol.

This scenario is an emulated one, and has been created using OpenLEON, which is a MEC topology emulator proposed in [22] which merges srsLTE [43] and Containernet [37] to deploy a scenario with a part of data-centre protocols and another part with Long Term Evolution (LTE) protocols. This emulator proposes a three-tier hierarchical topology, which means that the emulator deploys two core switches, having on each one two aggregation switches and sixty-four hosts connected. This topology allows the users to carry out test on an Edge architecture. In addition, thanks to the integration of srsLTE, it is possible to emulate a local mobile network and connect to the data-centre topology. The whole topology is managed by a Software Defined Network (SDN) controller that is implemented with RYU [41] which is a Python module. Its topology can be seen in Figure 7.

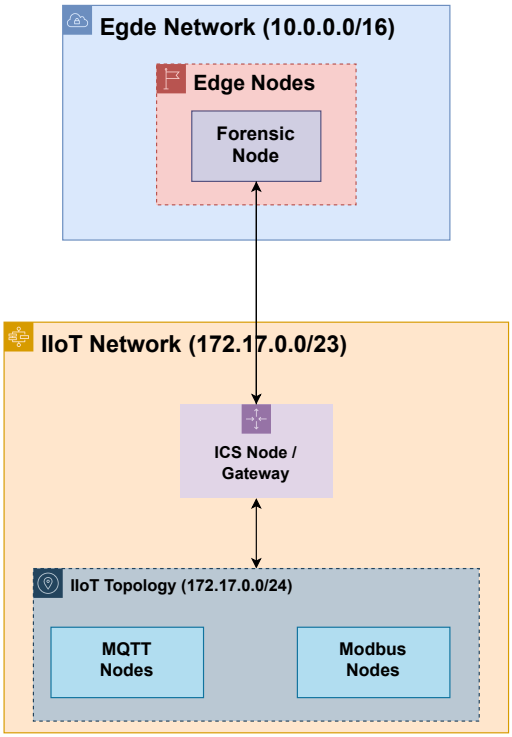

Figure 7: Second case study's topology.

With the goal of simulating the whole investigation process, meaning both the proactive and reactive phases and their interaction, apart from following the methodology, it is necessary to reproduce an incident in the environment that can lead to the initiation of a reactive process. Therefore, the actions carried out in this case study to have been the following:

- Creation and configuration of the IoT network and its devices.
- Launch of both the MQTT and Modbus services, simulating a normal behaviour.

- Insertion of the edge node into the network and start applying the proposed methodology.
- Generation of an anomaly and start of the reactive process.

Detection. Since there are two different protocols being used in the network, the edge node can be configured to work with both of them separately, together or just one of them. In this case study, the three options are presented, collecting traffic for each protocol individually and also for the whole network without filtering by protocol in order to show the adaptability of the proposal.

When the node is installed for the first time, it studies the whole traffic in the IoT network, and then provides information about the protocols being used and the active devices and their behaviour, as can be seen in Figure 8. Whenever the node performs a study of the network, the captured file used to evaluate the data is safely stored as well.

In this scenario, the edge node is working with three different sets of network traffic, to which the anomaly detection algorithm can be applied. As a result, the network administrator will receive an alert informing of the anomalies detected for each capture, and the specific connection which caused it. In Figure 9[1], a graphic example is presented showing the result of applying the anomaly detection algorithm in the traffic generated by one of the protocols, namely the MQTT service.

Pre-Process. Once that the incident has been detected, the investigator has all the information that has been gathered during the "Detection" phase. Therefore, they know the number of devices on the network, the protocols they use, their ID, and whether they can be remotely accessed, thus giving them plenty of valuable data that may be helpful for designing the action plan, as well as they have the possibility of executing any precautionary action if needed, since the edge node can interact with the devices in the network.

Identification. Using the same information extracted in the "Pre-Process" phase, the investigator can easily know the range of the investigation and determine which devices were active when the incident arose. Additionally, as the edge node tries to remotely connect to the devices in the network, it also stores information on the outcome of the connections, which ultimately means whether the device's storage is acquirable following a remote method. In this scenario, the only IoT unit which allows this type of connection is the central node. When checking the log stored by the node, which is shown in Figure 10[2] it can be seen that, first and foremost, the central node can be remotely accessed. Secondly, it determines whether the rest of devices are accessible as well, which they are not, therefore their file system cannot be acquired following a remote approach.

Acquisition & Preservation. When it comes to executing the acquisition process, since the central node is the only one that allows doing so remotely, the edge node connects to it and executes the "dd" command, obtaining a forensic image of the file system. In order to proceed with the acquisition, it lists the file systems available in the device, so that the investigator can select the one that they want to collect. The whole process is graphically shown in Figure 11.

This forensic image is stored in the edge node, and then a hash calculation is made, generating an acquisition log identical to the one showed for the network traffic.

Regarding the rest of the devices, since they are not physical devices, and they cannot be remotely accessed, there is no way of accessing the data that they store using a forensically sound technique (the data of the emulated devices can be accessed, but due to the openLEON emulator allowing to do so for managing purposes).

Analysis. In this case, since the investigator knows, due to the information provided in the "Identification" phase, that the central node can be remotely accessed, the possibility of performing a remote analysis exists. However, as mentioned in Section 3.5, this proposal does not automatize this process, only gives information on its feasibility.

On top of that, since the acquisition phase was successful, the investigator is able to carry out an offline analysis of the central node by studying the forensic image file collected by the edge node.

With respect to the sensors, only the network traffic can be studied, since there is no way of remotely accessing them, and the acquisition process was not feasible.

Therefore, after following this methodology, the investigator is able to analyze the following sources of evidence:

- The network traffic of the whole network.
- The MQTT traffic.
- The Modbus traffic.
- The file system of the central node.
- The data stored by the central node when it is operating, by remotely connecting to it.

[1] The anomalies are labelled as "-1" by the algorithm, that is why only the packets and connections with this value are shown

[2] In this case, the test results are saved as "1" if the device is remotely accessible, and as "0" if it is not

```
The protocol TCP has been detected with 192 packets exchanged

The protocol MQTT has been detected with 76 packets exchanged

The protocol Modbus/TCP has been detected with 181 packets exchanged

The following devices have been detected:

       Device
0  172.17.0.3
1  172.17.0.2
2  172.17.0.4
3  172.17.0.6
4  172.17.0.5

The number of packets sent by each device is:

            Packets Sent
Source
172.17.0.2          173
172.17.0.3          100
172.17.0.4           96
172.17.0.5           22
172.17.0.6           58

The number of packets received by each device is:

             Packets Received
Destination
172.17.0.2                276
172.17.0.3                 66
172.17.0.4                 65
172.17.0.5                 12
172.17.0.6                 30


The last packet sent by the device with IP 172.17.0.3 was in 2022-11-02 00:13:42.619244018 UTC Time
The last packet sent by the device with IP 172.17.0.2 was in 2022-11-02 00:13:42.737941166 UTC Time
The last packet sent by the device with IP 172.17.0.4 was in 2022-11-02 00:13:42.619419899 UTC Time
The last packet sent by the device with IP 172.17.0.6 was in 2022-11-02 00:13:42.725023755 UTC Time
The last packet sent by the device with IP 172.17.0.5 was in 2022-11-02 00:13:42.737954526 UTC Time

Acquisition started: 2022-11-02 00:13:10.494129335
Acquisition ended: 2022-11-02 00:13:42.737954526
File generated: Capture_2022-11-02 00:13:10.494129335.pcap
MD5 cheksum: 9c14cb47db3fdad07e10f6033b8bbe32
SHA1 checksum: f8737e53f247d6e48ffebe302baa5b824e0f3f87
```

Figure 8: Protocols detected by the edge node after studying the traffic in the IoT network.

	Source	Destination	Connections	Anomaly
0	172.17.0.2	172.17.0.3	80	-1
3	172.17.0.4	172.17.0.2	30	-1

(a) Anomaly detection at connection level.

	No.	Time	Source	Destination	Protocol	Length	Info	Anomaly
9	10	2022-11-02 23:51:19.855963528	172.17.0.2	172.17.0.3	MQTT	70	Connect Ack	-1
78	79	2022-11-02 23:51:39.358311664	172.17.0.2	172.17.0.4	MQTT	70	Connect Ack	-1
80	81	2022-11-02 23:51:39.358335064	172.17.0.4	172.17.0.2	MQTT	78	Subscribe Request (id=1) [topic]	-1

(b) Anomaly detection at packet level.

Figure 9: Result of applying the anomaly detection algorithm in the MQTT traffic.

Conclusions. The following conclusions can be drawn after performing this case study:

- The monitoring task can lead to detecting incidents almost instantly and, what it is most important, they are spotted based on the data generated by the devices. This is a clear advantage in terms of speed, since, in a conventional methodology, the usual procedure is for a human being to notify an abnormal behaviour of the system, which can be done days after the incident actually arose.
- It is possible to track the changes of devices in the network in terms of units being added, removed or no longer being used. This means that there is in-

```
The remote test result is the following:

       Device  Remote Access Test
0  172.17.0.3                   1
1  172.17.0.2                   0
2  172.17.0.4                   0
3  172.17.0.6                   0
4  172.17.0.5                   0


The device 172.17.0.3 can be remotely accessed through SSH
The device 172.17.0.2 cannot be remotely accessed
The device 172.17.0.4 cannot be remotely accessed
The device 172.17.0.6 cannot be remotely accessed
The device 172.17.0.5 cannot be remotely accessed
```

Figure 10: Information regarding the possibility of remotely accessing the IoT devices in the network.

```
Connecting to: 172.17.0.3

Connected to: 172.17.0.3

The following filesystems are available:

Filesystem     1K-blocks    Used Available Use% Mounted on

udev              393424       0    393424   0% /dev

tmpfs              92792    8552     84240  10% /run

/dev/mmcblk0p2  29699400 1120648  27057128   4% /writable

/dev/loop0         50176   50176         0 100% /

/dev/loop1        189696  189696         0 100% /lib/modules

/dev/mmcblk0p1    258095  128818    129278  50% /boot/uboot

tmpfs              92788       0     92788   0% /run/user/1000


Acquiring /dev/mmcblk0p1

524288+0 records in

524288+0 records out

268435456 bytes (268 MB, 256 MiB) copied, 16.0184 s, 16.8 MB/s


/dev/mmcblk0p1 acquired
```

Figure 11: Process of acquiring the file system from the central node.

formation available at any moment on which devices are actively having an impact in the network. Therefore, the identification process is done by studying the behaviour of the devices and not by physically locating them.

– The edge node has provided information on whether the devices can be remotely accessed, which is of value for both the acquisition and analysis phase.
– In the exact moment in which the anomaly is detected, an acquisition of the storage of the central node can be made. In addition, the investigator also has access to the network traffic, as it is constantly being collected.
– If wanted, the edge node can be configured to automatically perform acquisitions periodically, which can be extremely useful for the “Recovery” phase, as the user would have available several backups which can be used to bring back the system to a functioning state.

5 Answers to the Research Questions

After completing the experiment, and gathering a significant amount of knowledge in the design of automatic solutions for IoT forensic investigations, the research questions formulated in the beginning of this research can be answered.

- **(RQ1)** Given the lack of IoT-centered forensic tools, in which aspects, if any, can edge devices be useful to assist investigators in IoT examinations?
  - The use of an edge node allows having an intermediary between the IoT devices and the investigator, which is of relevance given the difficulty of accessing the former. This way, the examiner has a device in which they can execute forensic tasks, such as the evidence acquisition, that they know that is capable of doing so and is compatible with the tools needed to carry out the process. In addition, it facilitates the operation of interacting with the network traffic, as the edge node can be configured to be able to access the most common IoT protocols. Given the relevance that this source of evidence has in the IoT, this aspect can be crucial when performing a forensic investigation.
- **(RQ2)** Seeing the limitations that IoT models, methodologies, and frameworks have in terms of usability in a court of law as a result of their novelty, can the approach of adapting conventional solutions to the IoT's requirements meet the expected criteria when it comes to handling and preserving pieces of evidence?
  - It can, but following the conventional models when it comes to designing how to act in the acquisition and preservation phases of the investigation means compromising the effectiveness of the proposals. In the end, the procedure that is usually followed to preserve the collected evidence is to use hash codes to evaluate the integrity of the data, combined with following the chain of custody. This can be applied to any collected data. However, the validity of the evidence acquired is questioned when remote/live/online techniques are used, as it is difficult to estimate with certainty the impact that the actions executed by the investigator to retrieve the data have had in the data stored in the device. As a result, there is little room for flexibility on IoT investigations when physical techniques cannot be executed. An improvement is needed, not only in the techniques and tools available to carry out the acquisition and preservation of the pieces of evidence, but also in the way that investigators, and, most importantly, the legal sector perceive performing a remote technique and how to measure its impact in terms of evidence alteration.
- **(RQ3)** Considering the multiple approaches followed by the forensic community when it comes to identifying sources of evidence, such as network zone division, relying on logical communications, or using external solutions, in which aspects using an edge node could be an interesting way of tackling this phase?
  - Using an edge node is interesting as it provides a way of interacting with the network data generated by the IoT devices. Therefore, it is an approach that uses an external device, but also relies on logical communications to identify the devices. If used in a proactive mode, it can keep track of all the changes that have happened in the network, so there is a history available to the examiner on the devices that have been part of it. Similarly, when used in a reactive way, it can study the network traffic to determine the number of devices in the network, making it much easier to establish the range of the investigation.
- **(RQ4)** Performing the acquisition of IoT devices has proven to be a difficult task due to the impossibility in many cases of carrying out either a physical or remote technique. Taking into account that there are no IoT-centered procedures to collect the data generated by IoT devices, how can an investigator overcome this issue and therefore be able to perform an analysis of the devices under examination?
  - By relying on the network traffic as a source of evidence. There are two main reasons to do so. Firstly, the interoperability of the environment leads to the data being exchanged on-the-fly and, with the devices having little storage, there are less information to extract from the non-volatile memory. Therefore, the network traffic depicts a more accurate representation of the state of the IoT network and its devices. Secondly, not only the non-volatile memory is a difficult one to acquire, the incompatibility of volatile memory acquisition tools and profile creation ones with IoT devices makes RAM memory another challenging source of evidence to collect and analyze. As a result, the only remaining option for the examiner is to rely on network traffic to be able to study some data, so it must be treated with high importance.

6 Conclusions

In this article, we have addressed the design of IoT forensic methodologies. After evaluating the state of the field, it has been noticed that there is a lack of solutions

that, firstly, are IoT-centered, and, secondly, guarantee carrying out the investigations in a complete and efficient way. This is especially noticeable in two key aspects of the forensic process. The first one is when it comes to identifying the possible sources of evidence in a scene, which is usually known as the "Identification" phase. This is due to the dynamism of IoT networks and the high number of devices that operate in them, which makes it difficult to keep track of them. The second issue arises when the investigator wants to collect the data generated by IoT devices, namely the "Acquisition" phase. In this case, the field is limited by the existing techniques, which were designed to be used on conventional devices, and the time of life of the pieces of evidence, which is extremely low. This leads to physical acquisition techniques being highly incompatible and difficult to execute, making remote ones the most feasible ones, which are not so common on conventional scenarios, so they are not prioritized on the existing methodologies.

Upon inspecting the proposals from the research community, it can be seen that the solutions designed for IoT investigations are not capable of entirely solving the issues above-mentioned, and one of the main reasons why this happens is due to the lack of new techniques or devices that may change the manner in which an investigator interacts with the pieces of evidence. In this way, the use of edge computing has proven to be an interesting way of accessing the information that is exchanged in an IoT network without causing any interference. Articles focusing on other cybersecurity fields, such as attack or intrusion detection, have been proposed in which the results have been promising.

Under these circumstances, this research presents an IoT forensic methodology that uses the edge computing paradigm to assist in the investigation process. The resulting proposal is an eight phase methodology that covers the whole investigation process, from the design of the action plan to the recovery of the devices. Furthermore, it adds the possibility of helping in the detection of an incident, automatically launching the forensic process. By following this approach, it is possible to ensure a minimum level of completeness on IoT investigations. Regarding the "Identification" phase, it is guaranteed the detection of the active devices in the IoT network. With respect to the "Acquisition" it makes the collection of the network traffic feasible in any environment. This way, the investigator most certainly will have some sources of evidence available to study. Otherwise, they may not be able to study any relevant data.

In fact, when tested in two case studies representing incidents that could arise in real life, one focused on a smart home, and another in a smart industry, it can be seen that following this methodology leads to the effective identification, acquisition, preservation, and analysis of sources of evidence, addressing some of the existing issues on IoT forensic investigations. In addition, it also allows the reduction of the response time in case of an incident, thanks to the possibility of using this proposal in a proactive scenario.

6.1 Future Work

This work is a starting point for the integration of edge computing in the design of IoT forensic methodologies, therefore there are some interesting projects that derive from it that could be approached in the future:

- Carrying out additional tests in other IoT scenarios.
- Performing a step-by-step comparison in a practical scenario with other IoT methodologies, and even conventional ones, in order to find the advantages and disadvantages of using this proposal compared to others.
- Evaluating the impact that would have migrating the data storage and anomaly detection to the cloud.
- Studying further methods of safely storing the acquired sources of evidence, so that a higher level of preservation can be achieved.

Acknowledgements

This work has been developed with the help of Prof. Dr.-Ing. Felix Freiling from the Friedrich-Alexander-Universität Erlangen-Nürnberg.

Funding: This research was supported by the University of Castilla-La Mancha under the contract 2022-PRED-20677, and the project 2022-GRIN-34056, by the Spanish Ministry of Economic Affairs and Digital Transformation under the project PID2021-123627OB-C52, and by the Regional Government of Castilla-La Mancha under the project SBPLY/21/180501/000195.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.
Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.
Availability of data and material Not applicable.
Code availability Not applicable.

References

1. Github. zigbee_tools. a few zigbee tools to compliment killerbee (2016). URL https://github.com/inguardians/zigbee_tools/tree/master
2. Connectivity and Mobile Trends Survey (2022). URL https://www2.deloitte.com/us/en/pages/about-deloitte/articles/press-releases/connectivity-and-mobile-trends.html
3. Global IoT connections 2030, by application (2023). URL https://www.statista.com/statistics/1403256/global-iot-connections/
4. Scapy. a powerful interactive packet manipulation library written in python (2023). URL https://scapy.net/
5. Ai, Y., Peng, M., Zhang, K.: Edge Computing Technologies for Internet of Things: a Primer. Digital Communications and Networks **4**(2), 77–86 (2018). DOI https://doi.org/10.1016/j.dcan.2017.07.001
6. Al-Haj Baddar, S., Merlo, A., Migliardi, M.: Behavioral-Anomaly Detection in Forensics Analysis. IEEE Security Privacy **17**(1), 55–62 (2019). DOI 10.1109/MSEC.2019.2894917
7. Al-Masri, E., Bai, Y., Li, J.: A Fog-Based Digital Forensics Investigation Framework for IoT Systems. In: 2018 IEEE International Conference on Smart Cloud (SmartCloud), pp. 196–201 (2018). DOI 10.1109/SmartCloud.2018.00040
8. Alabdulsalam, S., Schaefer, K., Kechadi, T., Le-Khac, N.A.: Internet of Things Forensics – Challenges and a Case Study. In: G. Peterson, S. Shenoi (eds.) Advances in Digital Forensics XIV, pp. 35–48. Springer International Publishing, Cham (2018)
9. Almogren, A.S.: Intrusion detection in Edge-of-Things computing. Journal of Parallel and Distributed Computing **137**, 259–265 (2020). DOI https://doi.org/10.1016/j.jpdc.2019.12.008. URL https://www.sciencedirect.com/science/article/pii/S074373151930872X
10. Badenhop, C.W., Graham, S.R., Mullins, B.E., Mailloux, L.O.: Looking Under the Hood of Z-Wave: Volatile Memory Introspection for the ZW0301 Transceiver. ACM Trans. Cyber-Phys. Syst. **3**(2) (2018). DOI 10.1145/3285030. URL https://doi.org/10.1145/3285030
11. Badenhop, C.W., Ramsey, B.W., Mullins, B.E., Mailloux, L.O.: Extraction and analysis of non-volatile memory of the zw0301 module, a z-wave transceiver. Digital Investigation **17**, 14 – 27 (2016). DOI https://doi.org/10.1016/j.diin.2016.02.002. URL http://www.sciencedirect.com/science/article/pii/S1742287616300214
12. Bhuyan, M.H., Bhattacharyya, D.K., Kalita, J.K.: Network Anomaly Detection: Methods, Systems and Tools. IEEE Communications Surveys Tutorials **16**(1), 303–336 (2014). DOI 10.1109/SURV.2013.052213.00046
13. Canonical Group: Ubuntu Core - Ubuntu. https://ubuntu.com/core (2023)
14. Case, A., Richard, G.G.: Memory forensics: The path forward. Digital Investigation **20**, 23–33 (2017). DOI https://doi.org/10.1016/j.diin.2016.12.004. URL https://www.sciencedirect.com/science/article/pii/S1742287616301529. Special Issue on Volatile Memory Analysis
15. Castelo Gómez, J.M., Carrillo Mondéjar, J., Roldán Gómez, J., Martínez Martínez, J.L.: A context-centered methodology for IoT forensic investigations. International Journal of Information Security (2020). DOI 10.1007/s10207-020-00523-6. URL https://doi.org/10.1007/s10207-020-00523-6
16. Castelo Gómez, J.M., Carrillo-Mondéjar, J., Roldán-Gómez, J., Martínez Martínez, J.L.: A Concept Forensic Methodology For The Investigation Of IoT Cyberincidents. The Computer Journal p. bxad062 (2023). DOI 10.1093/comjnl/bxad062. URL https://doi.org/10.1093/comjnl/bxad062
17. Computer Hope: Linux and Unix dd Command. http://www.computerhope.com/unix/dd.htm (2023)
18. Cruz, T., Barrigas, J., Proença, J., Graziano, A., Panzieri, S., Lev, L., Simões, P.: Improving Network Security Monitoring for Industrial Control Systems. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 878–881 (2015). DOI 10.1109/INM.2015.7140399
19. DFRWS Attendees: A Road Map for Digital Forensic Research. Tech. rep., DFRWS (2010). URL https://dfrws.org/wp-content/uploads/2019/06/2001_USA_a_road_map_for_digital_forensic_research.pdf
20. Elstner, J., Roeloffs, M.: Forensic analysis of newer tomtom devices. Digital Investigation **16**, 29 – 37 (2016). DOI https://doi.org/10.1016/j.diin.2016.01.016. URL http://www.sciencedirect.com/science/article/pii/S174228761630010X
21. Eskandari, M., Janjua, Z.H., Vecchio, M., Antonelli, F.: Passban IDS: An Intelligent Anomaly-Based Intrusion Detection System for IoT Edge Devices. IEEE Internet of Things Journal **7**(8), 6882–6897 (2020). DOI 10.1109/JIOT.2020.2970501
22. Fiandrino, C., Pizarro, A., Mateo, P., Andrés Ramiro, C., Ludant, N., Widmer, J.: openLEON: An End-to-end Emulation Platform From the Edge Data Center to the Mobile User. Computer Communications **148**, 17–26 (2019). DOI 10.1016/j.comcom.2019.08.024
23. Forcier, J.: Welcome to Paramiko! — Paramiko documentation (2023). URL https://www.paramiko.org/
24. Gyamfi, E., Ansere, J.A., Xu, L.: ECC Based Lightweight Cybersecurity Solution For IoT Networks Utilising Multi-Access Mobile Edge Computing. In: 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC), pp. 149–154 (2019). DOI 10.1109/FMEC.2019.8795315
25. Hazra, A., Alkhayyat, A., Adhikari, M.: Blockchain-aided Integrated Edge Framework of Cybersecurity for Internet of Things. IEEE Consumer Electronics Magazine pp. 1–1 (2022). DOI 10.1109/MCE.2022.3141068
26. Hossain, M., Hasan, R., Zawoad, S.: Trust-IoV: A Trustworthy Forensic Investigation Framework for the Internet of Vehicles (IoV). In: 2017 IEEE International Congress on Internet of Things (ICIOT), pp. 25–32 (2017). DOI 10.1109/IEEE.ICIOT.2017.13
27. Itodo, C., Varlioglu, S., Elsayed, N.: Digital Forensics and Incident Response (DFIR) Challenges in IoT Platforms. In: 2021 4th International Conference on Information and Computer Technologies (ICICT), pp. 199–203 (2021). DOI 10.1109/ICICT52872.2021.00040
28. Kim, M., Shin, Y., Jo, W., Shon, T.: Digital Forensic Analysis of Intelligent and Smart IoT Devices. The Journal of Supercomputing **79**(1), 973–997 (2023). DOI 10.1007/s11227-022-04639-5. URL https://doi.org/10.1007/s11227-022-04639-5
29. Le-Khac, N.A., Jacobs, D., Nijhoff, J., Bertens, K., Choo, K.K.R.: Smart vehicle forensics: Challenges and case study. Future Generation Computer Systems (2018). DOI https://doi.org/10.1016/j.future.2018.05.081. URL

http://www.sciencedirect.com/science/article/pii/S0167739X17322422
30. Lionel Sujay Vailshery. Statista: IoT Connected Devices Worldwide 2019-2030 - Statista. https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/ (2023)
31. Maggi, F., Zanero, S., Iozzo, V.: Seeing the Invisible: Forensic Uses of Anomaly Detection and Machine Learning. SIGOPS Oper. Syst. Rev. **42**(3), 51–58 (2008). DOI 10.1145/1368506.1368514. URL https://doi.org/10.1145/1368506.1368514
32. Moteiv Corporation: Tmote sky Datasheet (2006). URL http://www.crew-project.eu/sites/default/files/tmote-sky-datasheet.pdf
33. Nieto, A., Rios, R., Lopez, J.: A Methodology for Privacy-Aware IoT-Forensics. In: 2017 IEEE Trustcom/BigDataSE/ICESS, pp. 626–633 (2017). DOI 10.1109/Trustcom/BigDataSE/ICESS.2017.293
34. Oriwoh, E., Jazani, D., Epiphaniou, G., Sant, P.: Internet of Things Forensics: Challenges and Approaches. In: 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, pp. 608–615 (2013). DOI 10.4108/icst.collaboratecom.2013.254159
35. Pan, J., Yang, Z.: Cybersecurity Challenges and Opportunities in the New "Edge Computing + IoT" World. In: Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization, SDN-NFV Sec'18, p. 29–32. Association for Computing Machinery, New York, NY, USA (2018). DOI 10.1145/3180465.3180470. URL https://doi.org/10.1145/3180465.3180470
36. Perumal, S., Norwawi, N.M., Raman, V.: Internet of Things(IoT) Digital Forensic Investigation Model: Top-down Forensic Approach Methodology. In: 2015 Fifth International Conference on Digital Information Processing and Communications (ICDIPC), pp. 19–23 (2015). DOI 10.1109/ICDIPC.2015.7323000
37. Peuster, M.: Containernet - Use Docker Containers as Hosts in Mininet Emulations (2023). URL https://containernet.github.io/
38. Qiu, T., Chi, J., Zhou, X., Ning, Z., Atiquzzaman, M., Wu, D.O.: Edge Computing in Industrial Internet of Things: Architecture, Advances and Challenges. IEEE Communications Surveys & Tutorials **22**(4), 2462–2488 (2020). DOI 10.1109/COMST.2020.3009103
39. Raspberry Pi Foundation: Buy a Raspberry Pi 3 Model B – Raspberry Pi. https://www.raspberrypi.org/products/raspberry-pi-3-model-b/ (2023)
40. Razaque, A., Aloqaily, M., Almiani, M., Jararweh, Y., Srivastava, G.: Efficient and reliable forensics using intelligent edge computing. Future Generation Computer Systems **118**, 230–239 (2021). DOI https://doi.org/10.1016/j.future.2021.01.012. URL https://www.sciencedirect.com/science/article/pii/S0167739X21000224
41. Ryu SDN Framework Community: Ryu SDN Framework (2023). URL https://ryu-sdn.org/
42. Simmons, A.: Internet of Things (IoT) Examples by Industry in 2023 (2023). URL https://dgtlinfra.com/internet-of-things-iot-examples/
43. Systems, S.R.: srsRAN - Your own mobile network (2023). URL https://www.srslte.com/
44. Texas Instrument Inc.: CC2531 System-on-Chip Solution for IEEE 802.15.4 and ZigBee Applications datasheet (Rev. A) (2010). URL https://www.ti.com/lit/ds/symlink/cc2531.pdf?ts=1694769897525&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FCC2531
45. Vural, I., Venter, H.: Mobile Botnet Detection Using Network Forensics. In: A.J. Berre, A. Gómez-Pérez, K. Tutschku, D. Fensel (eds.) Future Internet - FIS 2010, pp. 57–67. Springer Berlin Heidelberg, Berlin, Heidelberg (2010). DOI 10.1007/978-3-642-15877-3_7
46. Watson, S., Dehghantanha, A.: Digital forensics: the missing piece of the internet of things promise. Computer Fraud & Security **2016**(6), 5–8 (2016). DOI https://doi.org/10.1016/S1361-3723(15)30045-2. URL https://www.sciencedirect.com/science/article/pii/S1361372315300452
47. Wei, C., Xie, G., Diao, Z.: A Lightweight Deep Learning Framework for Botnet Detecting at the IoT Edge. Computers & Security p. 103195 (2023). DOI https://doi.org/10.1016/j.cose.2023.103195. URL https://www.sciencedirect.com/science/article/pii/S0167404823001050
48. Wurm, J., Hoang, K., Arias, O., Sadeghi, A., Jin, Y.: Security analysis on consumer and industrial iot devices. In: 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 519–524 (2016). DOI 10.1109/ASPDAC.2016.7428064
49. Xiaomi: Mi Global Home (2022). URL https://www.mi.com/global/mi-smart-sensor-set/. Accessed on 2022-07-19
50. Yu, W., Liang, F., He, X., Hatcher, W.G., Lu, C., Lin, J., Yang, X.: A Survey on the Edge Computing for the Internet of Things. IEEE Access **6**, 6900–6919 (2018). DOI 10.1109/ACCESS.2017.2778504
51. Yusoff, Y., Ismail, R., Hassan, Z.: Common Phases of Computer Forensics Investigation Models. International Journal of Computer Science & Information Technology (IJCSIT) **3** (2011). DOI 10.5121/ijcsit.2011.3302