Intrusion Detection at the Network Edge: Solutions, Limitations, and Future Directions

Simone Raponi(✉), Maurantonio Caprolu, and Roberto Di Pietro

College of Science and Engineering (CSE),
Division of Information and Computing Technology (ICT),
Hamad Bin Khalifa University (HBKU), Doha, Qatar
{sraponi,mcaprolu}@mail.hbku.edu.qa, rdipetro@hbku.edu.qa

Abstract. The low-latency, high bandwidth capabilities promised by 5G, together with the diffusion of applications that require high computing power and, again, low latency (such as videogames), are probably the main reasons—though not the only one—that have led to the introduction of a new network architecture: Fog Computing, that consists in moving the computation services geographically close to where computing is needed. This architectural shift moves security and privacy issues from the Cloud to the different layers of the Fog architecture. In this scenario, IDSs are still necessary, but they need to be contextualized in the new architecture. Indeed, while on the one hand Fog computing provides intrinsic benefits (e.g., low latency), on the other hand, it introduces new design challenges.

In this paper, we provide the following contributions: we analyze the possible IDS solutions that can be adopted within the different Fog computing tiers, together with their related deployment and design challenges; and, we propose some promising future directions, by taking into account the challenges left uncovered by the considered solutions.

1 Introduction

The data deluge expected by the massive adoption of IoT solutions, together with the need for better network performance required by modern end-user applications, underline how the classic network Cloud model is not able to efficiently respond to the new needs. The Cloud model offers a scalable infrastructure that frees users from the costs of designing, purchasing, and maintaining computing and storage resources. Despite the obvious advantages, this model is not suitable for latency sensitive applications, that demand geographical proximity with the service provider to meet their delay requirements. To address this challenge, Cisco researchers defined a new network architecture, called Fog Computing [1], that extends the Cloud computing paradigm to the edge of the network, enabling a new variety of applications and services, such as gaming, augmented reality, and real-time video stream processing. This new paradigm provides computational

T. Zhang et al. (Eds.): EDGE 2019, LNCS 11520, pp. 59–75, 2019.
https://doi.org/10.1007/978-3-030-23374-7_5

and storage capabilities physically closer to end-users, where data are being generated [2]. Among the characteristics of Fog Computing, the most important are [1]: low latency and location awareness; handling of a huge number of nodes; heterogeneity; widespread geographical distribution; support for mobile end-devices; support for real-time applications; and wireless access.

Since the Fog computing network architecture brings the typical services offered by Cloud computing closer to the end-user, most of its security and privacy issues are inherited from the Cloud itself. These problems include, but are not limited to, Distributed Denial of Service (DDoS) attacks, Man in the Middle (MitM) attacks, rogue gateway attacks, privacy leakage, privilege escalation attacks, service manipulation attacks, and injection of information. However, although the problems are the same in Fog computing, they should be contextualized in the new physical and logical elements of the Fog computing network architecture [3].

One of the most effective methods to solve the above-cited problems is the adoption of an Intrusion Detection System (IDS) to monitor and analyze the network traffic and the devices' behavior. Nevertheless, even IDSs need to be contextualized to the new network architecture. Indeed, designing an effective IDS requires to choose not only the IDS typology (e.g., Host-based IDS, Network-based IDS) and the methods of detection (e.g., anomaly-based IDS, signature-based IDS), but also the tier in the Fog computing architecture where to place it. Since the Fog computing network architecture is composed of three tiers, the placement of an IDS within a tier with respect to the others would completely change its capabilities.

Although the implementation of IDSs within the Fog Computing network architecture poses many challenges, whether inherited from the Cloud architecture or not, the introduced benefits could make the difference in certain scenario (e.g., the detection time plays a crucial role in defending a critical infrastructure).

Contributions: In this paper, we first provide an in-depth analysis of the IDSs implementation within the Fog computing network architecture by both identifying several design and deployment challenges inherited by the Cloud environment, and proposing new original ones. Further, we provide a detailed overview of a selected set of existing solutions. Among the proposed IDS solutions in the literature we considered both the ones specifically implemented for the Fog computing network architecture and the generic ones that have not been thought for the Fog paradigm—though they could be adopted within one or more Fog tiers (e.g., IDS for IoT devices that could be deployed in edge devices, IDS for Cloud). Moreover, we have mapped each existing solution to the challenges identified during the analysis, highlighting how none of the current solutions is able to satisfy most of them. Finally, we propose some future directions, taking into consideration the challenges left uncovered by the analyzed solutions.

Road-Map: The paper is organized as follows. In Sect. 2, we provide a technical background of the Fog computing network architecture. In Sect. 3, we study advantages and drawbacks that possible implementations of IDS in the Fog computing network architecture would bring. In Sect. 4, we provide an analysis of

the main challenges related to architectural design and deployment of IDS in the Fog computing network architecture. The description of the existing solutions is performed in Sect. 5, together with the related mapping to the challenges previously identified. In Sect. 6, we discuss the results and propose some future directions, while in Sect. 7 we report some concluding remarks.

2 Background

In this section, an overview of both Edge and Fog Computing is provided, together with their differences.

Although apparently similar and often interchangeably used, Edge Computing and Fog Computing present key differences that are not always easy to catch. Both the network architectures share the same main objective: bringing the computation closer to the user, thus reducing the network congestion and the end-to-end delay. As highlighted in [4], the differences concern:

1. *how data are handled*. How to process and analyze data gathered locally or received by other devices in the network;
2. *where to process data*. Where to put both intelligence and computing power. The common architecture is composed of several tiers, the choice of the intelligence and computing power placement is crucial and strongly dependent on the purpose.

In Edge Computing, each end-device plays an important active role in processing data locally rather than delegate it to the Cloud [4]. As a consequence, every device, being it a sensor, an actuator, or a network device, relies on its own computational power and storage resources to perform operations on data. The product of this analysis could be maintained locally in case the device is autonomous and able to take advantage of this information, otherwise, it is delivered to other upper-tiers devices, that are usually responsible for both the management of the device itself and other devices belonging to the same subnetwork. On the contrary, in Fog Computing, processing power and storage resources needed to process and analyze data collected from IoT devices are integrated into other devices that, in turn, are moved geographically closer to the data collection. Usually, the devices in question are network ones, placed only a few hops away from the edge devices [5].

2.1 Fog Computing

Figure 1 depicts one of the most widely adopted architectures in Fog Computing: the Three-Tier Fog computing architecture [4].

Tier 1 – Edge Devices. Tier 1 usually consists of Internet of Things devices, including sensors (e.g., temperature, proximity, pressure, chemical, motion detection, optical), actuators (e.g., chemical, power generation, pumps, valves), and

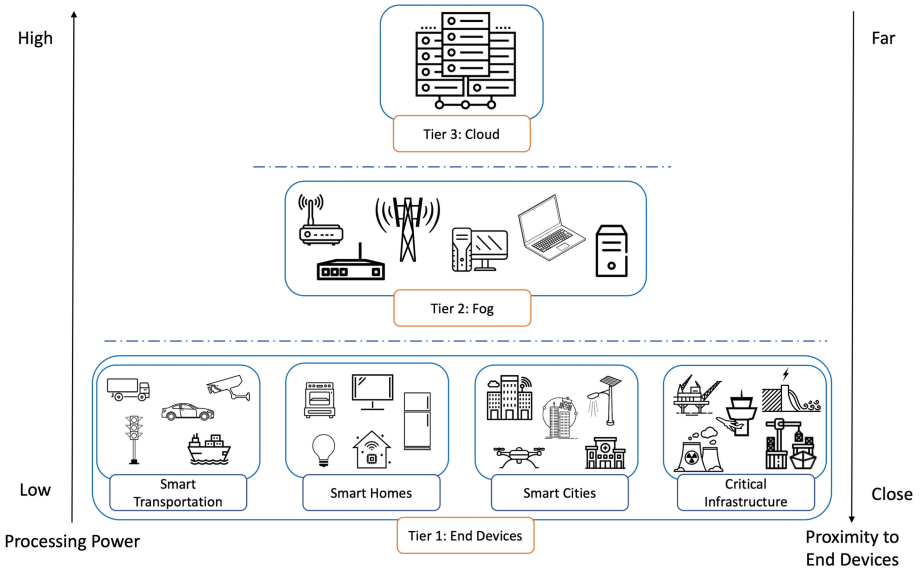

Fig. 1. Fog computing tiers

smart hand-held devices (e.g., smartphones, tablets, smart watches). Being the farthest from the Cloud tier, tier 1 includes devices that directly interact both with end-users and the surrounding environment, gathering data and information that need to be processed. However, most of these devices suffer from limited memory and limited computational power, thus being unable to apply algorithms for the analysis and the processing of the data in a limited time [6]. This limitation leads to the need to move the computation to more suitable places: in the Cloud, or within the tier 2 in case Fog computing network architectures have been adopted.

Tier 2 – Fog Devices. Tier 2 represents the layer between the end-user devices and the Cloud. It usually consists of network devices (including switches, routers, gateways), WAN/MAN (Wide Area Network/Metropolitan Area Network) access devices, multiplexers, Integrated Access Devices, and requires considerable resource requirements to perform several tasks, including real-time data processing and analysis, data storing, data caching, and computational offloading [7]. In this layer, the analysis performed on the gathered data obtained from the edge devices allows to take decisions locally, thus providing quick responses to unusual behaviors [8].

Tier 3 – Cloud. Tier 3 comprises the traditional Cloud servers. Cloud servers, for savings and efficiency reasons, are kept in dedicated facilities (i.e., data centers) that, in turn, are placed in convenient geographical locations (e.g., where

electricity is cheaper and the weather conditions are optimal). This leads to an unavoidable physical distance between the Cloud server and end-users, that eventually brings to end-to-end communication delays and network congestion. However, Cloud servers offer more computing power and more data storage with respect to devices in tier 2, together with the opportunity to perform computationally burdensome operations, such as management of big data and parallel data processing.

3 Intrusion Detection at the Network Edge

As a new in-standardization-phase network architecture, Fog Computing presents several security and privacy challenges. Indeed, although the Cloud architecture is commonly protected by Cloud operators, security and privacy solutions are not effortlessly extensible for the new architectures introduced. Challenges of these architectures include, but are not limited to, malicious insider attacks detection, malicious node detection, Fog forensics, intra-tier and inter-tier secure communication (i.e., authentication and integrity of the data exchanged), trust in the Fog services, trust in the end-users, cross-border privacy of data, security in the storage of the data, security and privacy in the computation of the data, and access control policies [4,9]. Since the Fog computing network architecture is continuously evolving, these challenges have only been partially addressed in the literature.

This section focuses on the detection of malicious attacks performed by internal/external attackers, by describing the possible implementations of IDSs in both the Cloud architecture and the Fog Computing network architecture. Attacks can come both from inside and outside the network. We consider:

- *outside attack*: any attack coming from outside the network or performed by a non-authenticated device;
- *inside attack*: any attack coming from inside the network and performed by an authenticated device.

Inside attacks are more difficult to detect, especially in multi-tenant environments where resources are shared among different applications and users. Moreover, attackers from the inside could use their knowledge of the system to cover their tracks, thus remaining transparent.

The Fog computing network architecture presents some peculiarities with respect to the Cloud architecture that could make an IDS more suitable and effective. The Fog computing network architecture is composed of three tiers, each tier offering a different view of the network: the higher the tier, the wider the vision on the network; conversely, the lower the tier, the narrower the vision on the network.

Advantages and drawbacks of implementing an IDS in the Fog computing network architecture with respect to the Cloud computing network architecture are depicted in Table 1. Each row in the table represents where it could be possible to deploy an IDS, while in each column a Fog computing feature is reported.

Table 1. Features of IDSs implemented in different network architectures.

		Comp. power	Storage	Bandwidth	Latency	Network view	Privacy threat
Cloud		High	High	High	High	High	High
Fog	Tier 3	High	High	High	High	High	High
	Tier 2	Medium	Medium	Medium	Low	Medium	Medium
	Tier 1	Low	Low	N/A	N/A	Low	Low

3.1 Implementing an IDS in the Cloud Servers

Implementing an IDS in the Cloud Servers or, equivalently, in Tier 3 of the Fog computing network architecture, allows to exploit the powerful resources of the Cloud devices, thus providing the IDS with remarkable *Computational Power* and *Storage* resources. Furthermore, the physical characteristics of the Cloud allow application and services to rely on an efficient and performing network, which guarantees an outstanding *Bandwidth*. However, the modern needs for a smart and connected world lead to a massive generation of data from edge devices spread all over the world. Considering that Cloud Servers are usually physically far from the edge devices, implementing an IDS on a Cloud Server implies high *latency* and delay times, inescapably dictated by the physical distance between the communicating devices. Being virtually placed at the highest point of the network, Cloud Servers have a far-reaching *Network View*, made available by devices located in lower levels that share information regarding their local view of the network. This gives an IDS the possibility to manage threats having a complete picture of the network available. Finally, an IDS has the ability to monitor data traffic and scan files in order to detect malicious code and unusual behaviors. Cloud service providers, to make the most of the computational and storage resources they own, rely on multi-tenant environments where users share the same machine, often without being aware of it. Although this approach is extremely cost-effective from the resources optimization's point of view, it opens the door to potentially serious privacy issues. In fact, regardless of the technology used (e.g., virtual machines, containers), there is an impressive amount of attacks aimed at undermining security and privacy of information on the Cloud [10].

3.2 Implementing an IDS in the Fog Network Architecture

The Fog computing network architecture, as depicted in Fig. 1, consists of three tiers, where each tier is composed of a specific set of devices: tier 1 includes edge devices, representing those devices that allow generating data, such as sensors, actuators, and smart-hand devices; tier 2 includes Fog devices, that are usually placed geographically close to the edge devices and perform processing and analysis services; tier 3 represents the Cloud itself. The advantages and drawbacks of an IDS in the Fog computing network architecture are strongly dependent on the tier in which it is implemented.

Implementing an IDS within the tier 3 makes no difference compared to implementing it within the Cloud since Fog computing network architecture's tier 3 and Cloud coincide. Conversely, the implementation of an IDS within the tier 2 presents some differences. Tier 2's devices, often represented by network devices (i.e., switches, routers, gateways), small servers, Integrated Access Devices, can boast of adequate *computational power* and *storage* resources, not even remotely comparable to those in the Cloud but sufficient to successfully perform data mining, data aggregation, and data processing tasks. Implementing an IDS within an existing network device would reduce the cost of acquisition, commissioning, and maintenance of a new device, with the disadvantage of subtracting computational resources from an operating network device. Furthermore, tier's 2 network devices could be not able to perform Intrusion Detection activities and manage the incoming traffic peaks in parallel, thus incurring in bottlenecks that could reduce the response time, and so the usefulness, of the IDS. The IDS can even be implemented within edge devices, placed at tier 1. In this latter case, considering that devices have poor *computational power* and *storage* resources, only lightweight IDSs (e.g., [11–13]) can be taken into account. Moreover, the Cloud's tier makes use of dedicated network backbones designed to handle a worldwide traffic volume, while tier 2's devices can only offer a limited *bandwidth*, having been designed to handle local traffic volumes. The bandwidth metrics do not make sense if we consider a tier 1's IDS, given that it would be directly integrated within the end-device. The *latency* is the metric that motivates the need to have IDSs within tier 2's devices. In fact, the geographic distance between Cloud servers and edge devices generating the data increases the *latency* in a perceptible way, an increase that could be critical in certain contexts (e.g., in a critical infrastructure scenario taking timely decisions is crucial, not respecting the right timing could lead to catastrophic events). The *Network View* becomes wider and wider with the increase of the tier, that is with the increase of the geographical distance from the edge devices. In fact, an IDS implemented within a tier 1's edge device enjoys only a limited vision of the network, that is the vision of the component itself (or of a small subset of them). Turning to tier 2, the devices receive data from subnetworks composed of edge devices, thus having the ability to both correlate and aggregate information, obtaining a wider vision of the network. Tier 3 exploits the information of tier 2's devices usually located in geographically distant places, that provide it with a peripheral view of the whole network. On the contrary, moving towards higher tiers, the threats related to the *privacy* of data and information tend to increase. In fact, considering that optimizing the use of resources leads to a maximization of the gain, the devices tend to serve as many clients as possible (and to manage as many devices of the lower tier as possible). The more information (often also important and sensitive) a device contains, the more this device is tempting for an attacker who, be it internal or external, will have at her disposal

various means to appropriate or compromise information. For example, a malicious employee could inject malware into a competitor company's network and mask the traces, compromising the information of a device that is dealing with the detection.

4 Challenges

In this section, we provide an in-depth analysis of the main challenges related to the architectural design and deployment of an IDS within the Fog. The new architecture introduced by the Fog computing network paradigm brings the typical services offered by Cloud computing closer to the end-user. For this reason, most of the security and privacy issues of the Fog computing architecture are inherited from the Cloud. Similar considerations can be applied to IDSs deployed within a Fog computing network architecture, that mainly have the same IDSs challenges as Cloud environments, in addition to new ones derived from IoT environments. We discuss all these challenges in the following subsections, starting from some considerations on the deployment of IDSs in the Fog computing network architecture environment that lead to some generally valid challenges, regardless of the type of IDS and the layer of the Fog computing network architecture in which it is deployed. Then, we present other challenges that arise depending on the type of IDS considered (Table 2).

4.1 Deployment

One of the first architectural problem to be considered before designing an IDS for the Fog computing network architecture environment is to define in which of the three tiers the single components of the system should be deployed. IDSs can be deployed within the tier 1 to detect malicious behavior by monitoring and analyzing log files, user login information, and enforcing access control policies. IDSs can also be deployed within the tier 2 to detect malicious attacks such as Denial of Service (DoS), port scanning, to name a few [9]. In order to increase the security level of the entire network, IDSs must be deployed in all the three tiers, monitoring and analyzing traffic and behavior of edge devices, Fog devices, and Cloud servers. In fact, securing one or two tiers of the Fog computing network architecture is not sufficient to protect the entire system, dangerous events like the propagation of malware from a compromised device to the rest of the network could not be noticed [4]. The deployment of IDS solutions within every tier of the Fog computing network architecture leads to common challenges, described in Sect. 4.2, as well as to specific ones depending on the type of IDS considered, discussed in Sect. 4.3.

4.2 General Challenges

Considering the Fog computing network architecture peculiarities described in Sect. 3, implementing a reliable and efficient IDS implies designing and tuning a detection system able to effectively work in an environment with the following characteristics:

- **Large-scale Network:** The large number of heterogeneous connected devices, as well as the unpredictable chaotic dynamics of today's large and medium-size computer networks, make the number of suspicious events that need to be controlled by an IDS huge. For this reason, IDSs should both be equipped with hardware resources adequate to support this workload and implement powerful algorithms to efficiently perform the required tasks.
- **Geo-distributed Environment:** In sharp contrast with the more centralized Cloud, the Fog could be very complex and geographically worldwide distributed, depending on the purpose for which it was designed [1]. As an example, we can consider a wireless sensor network deployed along a highway that crosses an entire country, providing lighting and video surveillance services. In this case, edge devices could be deployed every 100 m, while Fog devices could be deployed every kilometer. All these devices could send data to a Cloud located halfway around the world. In this scenario, an IDS should be able to provide a real-time protection to the entire architecture.
- **Real-time Notification:** The most important characteristic of every IDS is the ability to early discover intrusion violation threats. The huge number of connected edge devices, as well as their geographic distances (that have a severe impact on the network latency), make it difficult to analyze packets in real-time. This increases the notification time, impacting negatively on the response time.
- **Alarm Parallelization:** Securing all the Fog computing network architecture layers requires a distributed IDS system, with at least one component in every layer of the architecture. These components need to cooperate with each other by exchanging data that can be aggregated to obtain a comprehensive high-level view of the network, improving the overall reliability and reactiveness of the whole system.
- **False-alarm Control:** The main objective of an IDS is to raise alarm if an event in the network could be considered as an intrusion. The verification of whether a suspicious event is a real attack or a false positive is beyond the scope of current IDS solutions [14]. The Fog computing characteristics, discussed in Sect. 1, could increase the false positive/negative events detected by the system. For this reason, more efforts are needed to improve the IDSs detection accuracy.

Table 2. IDS design challenges

IDS type	Challenges
Host-Based Intrusion Detection System (Tier 1)	• Limited Resources (Comp. Power, Storage, Battery) • Lack of Context Knowledge • Delay in Centralized Reporting
Host-Based Intrusion Detection System (Tier 2)	• Lack of Context Knowledge • Delay in Centralized Reporting
Network-Based Intrusion Detection System (Tiers 1, 2, 3)	• Insider Attackers Detection • Cooperation • Decrease False Positives/Negatives • Encrypted Traffic • Developing Physical Jamming Detection Techniques • Developing DoS Detection and Mitigation Techniques
General	• Large-Scaling • Geo-Distribution • Environment Dynamicity • Real-time Notification • Alarm Parallelization • False-alarm Control

4.3 Design

The design challenges are different depending on the type of IDS considered (*Host-based* or *Network-based*) and the tier in which it is intended to be placed. The main challenges of deploying *host-based* IDS in the Fog computing network architecture are aligned with the ones of other network architectures:

- **Lack of Context Knowledge:** Having only its local view, a host-based IDS is not aware of what is happening outside. This lack of context knowledge makes more challenging to achieve high detection accuracy.
- **Delay in Centralized Reporting:** As part of a more complex system charged with supervising the whole Fog computing network, a host-based IDS has to report every detected local anomaly to a centralized entity. This entity is charged with the collection and elaboration of the data coming from every other component in the network. If the communication between this centralized module and the other components of the system has high latency due to their geographical distribution, the delay in centralizing reporting becomes very challenging, impacting the overall performance and accuracy of the IDS.

If the IDS is deployed within the tier 1, another challenge arises:

- **Limited Resources:** Since edge devices usually have very limited resources (i.e., in terms of computation, network bandwidth, storage, and battery), designing and implementing IDSs within the tier 1 could be very challenging.

For *network-based* IDSs instead, the main challenges are applicable to each tier of the Fog computing network architecture:

- **Insider Attackers Detection:** The attacks coming from inside the network are usually very challenging to discover. In fact, edge devices with genuine authentication privileges are often able to cover their traces and hide evidence of their malicious activities.
- **Cooperation:** If the IDS solution includes different modules, regardless of the tasks they are performing, they still need to cooperate with each other, adding further typical distributed systems' challenges.
- **Decrease False Positives/Negatives:** In the context of IDS, a high number of false positives makes the solution unusable due to the waste of resources dedicated to analyzing legitimate events. Moreover, false negatives make the solution ineffective, because malicious events would go unnoticed.
- **Encrypted Traffic:** A network-based IDS, due to its physical position, is able to observe the entire network traffic generated by the subnet it is connected to. However, if the traffic is encrypted, it is not able to open the packets and analyze their content. This limitation makes the detection of malicious packets more challenging.
- **Developing DoS Detection and Mitigation Techniques:** The Fog computing network architecture moves services from the Cloud to local Fog devices which, having less network bandwidth and being less protected, are more vulnerable to DoS attacks.

If the IDS is a Wireless IDS (WIDS), also the following challenge arises:

- **Developing Physical Jamming Detection Techniques:** Tier 1 is mostly composed of IoT sensor networks, which normally communicate both with each other and with Fog devices via wireless networks. This makes them vulnerable to physical level's DoS attacks, such as jamming attacks.

5 Existing Solutions

In this section we provide a thorough analysis of the studies related to IDSs present in the literature.

Given the widespread adoption of the Fog computing network architecture, in recent years several studies have come to light, with the aim of proposing solutions for the integration of efficient IDSs within the new paradigm. In [8], the authors introduced a lightweight IDS based on an Artificial Immune System (AIS), that is a form of biologically inspired computing. The AIS takes inspiration from the Human Immune System (HIS), that protects the body against the diseases being able to recognize external pathogens among internal cells and molecules of the body. The proposed IDS is developed in all the three tiers of the Fog computing network architecture. In tier 1, detectors are deployed within the edge devices. In tier 2, devices take advantage of a smart data concept to analyze and process the intrusion alerts. Smart data is a smart structure that helps to

manage a large amount of data in IoT applications: a simple lightweight data cell that evolves (by merging with other cells or dividing by them, according to the direction) when traveling across the tiers. Finally, in tier 3 the IDS organizes the network traffic in clusters (by relying on unsupervised clustering techniques) and trains the detectors.

In [15] the authors found in the Device security, thus in the identification of malicious edge devices, one of the major challenges for successfully integrate Fog Computing and Internet of Things. Taking into account the difficulty of preventing attacks from malicious Fog devices, due to their privileges of storing and processing data, the authors proposed a framework that makes use of three distinctive technologies: a two-stage Markov Model, an IDS, and a Virtual Honeypot Device (VHD). The two-stage Markov Model allows reducing the false alarm rate generated by the different types of data sent by the IoT devices. In detail, when the anomaly-based IDS detects a malicious behavior on the Fog device an attack alarm is generated and sent to the two-stage Markov Model. The first stage allows categorizing the Fog devices, while the second stage is dedicated to predicting whether the categorized devices have to be moved to the VHD or not. The VHD allows to store and maintain a log repository of all the identified malicious Fog devices and provides the system with protection against unknown attacks.

Furthermore, considering that the Fog computing network architecture provides the sensors networks with ever-increasing importance, several studies have introduced proposals of IDSs implementation within the aforementioned resource-constrained devices. In [16], the authors introduced a lightweight IDS based on a vector space representation using a single hidden layer MultiLayer Perceptron (MLP) to improve the detection time. The authors exploited new datasets, the Australian Defense Force Academy Linux Dataset (ADFA-LD) and the Australian Defense Force Academy Windows Dataset (ADFA-WD), respectively, representing system calls datasets containing both attacks and exploits on various applications. The proposed IDS, implemented within a Raspberry Pi as a Fog device, achieves 94% accuracy, 95% recall, and 92% F1-measure in ADFA-LD, and 74% accuracy, 74% recall, and 74% F1-measure in ADFA-WD when considering a small number of nodes. Another IDS able to run within resource-constrained devices has been introduced in [17]. The authors reached a convenient trade-off between the energy consumption and the accuracy detection by making use of an anomaly-based IDS only when the signature of a new attack, identified by a signature-based IDS, is expected to occur. The problem is formulated as a security game model, where the security strategy is a game formulation between the intruder's attack and the IDS agent implemented within an Internet of Things device. The IDS agent implements its anomaly detection techniques to detect new attack patterns by relying on the Nash Equilibrium. The performance and the viability of the proposed approach have been analyzed by simulating a Wireless Sensor Network (WSN) using the TOSSIM simulator.

However, at the top of the new network architecture, the Cloud continues to be omnipresent, so more and more innovative studies have been proposed with

the goal of implementing IDSs within the Cloud (or within the Fog computing network architecture's tier 3). In [18], the authors proposed an anomaly detection system at the hypervisor layer that makes use of Hybrid algorithms (e.g., Fuzzy C-Means clustering techniques, Artificial Neural Networks) to improve the accuracy of the detection system. The proposal has been experimented by using the DARPA's Knowledge Discovery and Data Mining (KDD) cup dataset, showing a higher detection accuracy and a lower false alarm rate even against low-frequent attacks, thus outperforming Naive Bayes classifiers and Classic ANN techniques.

In [19], the authors introduced a framework of Cooperative IDSs to counteract Distributed Denial of Service (DDoS) attacks on the Cloud. IDSs placed in the Cloud computing regions exchange alerts with each other. Each of them relies on a cooperative agent that is able to determine whether to accept the alert received from other IDSs or not. If the agent decides to accept the alert, the system adds a new blocking rule (related to the identification of the type of packet in the Cloud region) into the block table. A comparison against a pure Snort-based IDS shows that the proposed solution allows more accurate detection of Distributed Denial of Service attacks, paying only a small additional computational effort.

6 Discussion and Future Directions

Nowadays, several systems such as SCADA, Cloud, and Smart Grid rely on IDSs as the first line of defense against malicious attacks such as Scanning attacks, DoS attacks, Insider attacks, and Man in the Middle attacks [4]. For this reason, after the introduction of the Fog computing network architecture, a new line of research started studying the adoption of IDSs within this paradigm. Since the advantages of each IDS are strongly dependent on the tiers in which it is implemented, to increase the level of security, the IDSs should be deployed in every tier of the architecture. However, this choice brings new challenges, discussed in Sect. 4.

In this section, we first evaluate the mapping between existing solutions and these challenges, highlighting which challenges have not been addressed by the solutions in the literature, then we propose promising future directions.

Table 3 shows the mapping between the existing solutions in the literature and the challenges we identified during our analysis. A horizontal cut of the table allows to know whether the challenge has been addressed by the considered work, while a vertical cut provides an overview of the challenges addressed by single solutions. It is possible to notice how most of the solutions in the literature focused on solving the typical challenges of distributed systems (e.g., geographic distribution, large-scaling, environmental dynamicity, real-time notification, alarm parallelization, and delay in centralized reporting). This is justified by the fact that these solutions aim at leveraging the most important advantage offered by the Fog computing network architecture (i.e., the reduction of the network latency). This property allows Edge devices within the tier 1,

Table 3. Mapping between existing solutions and challenges *Legend: N/D: Not Declared, N/A: Not Applicable*

Challenges/Studies	[8]	[15]	[16]	[17]	[18]	[19]
Limited Resources	*N/D*	*N/A*	✗	✓	✗	✗
Lack of Context Knowledge	✓	✗	✗	✗	✓	✓
Delay in Centralized Reporting	✓	✓	✓	✗	✓	✓
Insider Attack Detection	✓	✗	✗	✓	✗	✗
Cooperation	✓	✗	✗	*N/A*	*N/A*	✓
Decrease False Positives/Negatives	✗	✓	✗	✓	✗	✓
Encrypted Traffic	✗	✗	*N/A*	*N/A*	✗	✗
Jamming Detection	✗	✗	✗	✗	*N/A*	*N/A*
DoS Detection	✓	✗	✗	✗	✓	✓
Large-Scaling	✓	✓	✓	*N/A*	✓	✓
Geographic Distribution	✓	✗	✓	*N/A*	✓	✓
Environment Dynamicity	✗	✓	✓	*N/A*	✓	✓
Real-time Notification	✓	✓	✓	✓	✓	✓
Alarm Parallelization	✓	✓	✓	*N/A*	*N/A*	✓
False-alarm Control	✗	✓	✗	✗	✗	✓

to quickly communicate with Fog devices within the tier 2, enabling more immediate aggregation and processing of data. In the context of IDS, this translates into improving the overall detection times of malicious events in the system.

However, most of the solutions did not focus on solving other important challenges, such as the development of lightweight IDSs able to work within resource-constrained devices, the false-alarm control, the reduction of false positive/negative number, and the DoS attack protection.

Tier 1 is typically composed of resource-constrained devices, with a limited amount of computational power, storage, and energy. These restrictions make the deployment of IDSs solutions within this tier challenging. In [17], for example, the authors proposed a lightweight detection technique that requires low energy consumption to achieve a high-security level.

Regarding the false-alarm control challenge, we believe that every IDS should have a validation mechanism for those events that are considered malicious, with the goal of decreasing the number of false positives. A possible solution requires to use more than one IDS's method of detection (i.e., signature-based, anomaly-based), that would also reduce the number of false negatives. In the context of IDSs, reducing false positives and false negatives is crucial, since a high number of false positives makes the solution unusable due to the waste of resources dedicated to analyzing legitimate events (that would be infeasible if the IDS has been deployed in a resource-constrained device), while false negatives make the solution ineffective, because malicious events would go unnoticed. Authors in [15] introduced a two-stage Markov module that helps to reduce the false-alarm rate of the IDSs.

The goal of some critical attacks on the Fog is to limit or deny the system services accessible to legitimate users/devices through Denial of Service attacks. In addition to the classic DoS attacks present in the literature, Edge devices could be infected by stealthy malware, that would consume their resources, finally leading to alternative DoS attacks. Although several solutions have been designed [20,21], this research field is still worthy of attention, and further contributions are needed to effectively face this challenge.

Table 3 also highlights that the existing solutions we took into account do not adequately respond to the encrypted traffic challenge. This stems from the fact that most of the solutions designed to work in the presence of encrypted traffic are limited to the detection of some specific types of attacks, such as scanning, brute-force, and DoS attacks, and are ineffective for all the others [22]. Advanced machine and deep learning techniques, together with deep packet inspection methods, could be integrated within an IDS with the goal of analyzing encrypted traffic to detect malicious patterns.

Another important future direction regards the integration of some jamming detection techniques on IDSs deployed within both tier 1 and 2 of the Fog computing network architecture. This would allow to detect jamming attacks and to react by putting in place specific countermeasures. For example, if we take into account a wireless sensor network deployed within the tier 1 that communicates with a Fog gateway (placed within the tier 2), a malicious user could be able to completely block the inter-tiers communication by jamming the wireless channel. One possible detection approach, deployed within the tier 2, involves the monitoring of the packet's throughput. A drastic fall of this parameter for one or more Edge devices detected by a Fog device could be strong evidence of malicious jamming activities. The detection of this attack is crucial because techniques aimed at restoring the communication could be implemented, such as relying on alternatives schemes [23].

It is worth mentioning that most of the challenges identified in Sect. 4 could be addressed by using SDN networks technologies, as highlighted by [24], that investigated the possible cooperation between Edge computing and SDN. In this field, a promising research direction is the implementation of security mechanisms using SDN switches with stateful data plane [25] within the tier 2 of the Fog architecture.

7 Conclusions

In this paper, we analyzed how the changes in the network architecture introduced by the adoption of Fog Computing affect both the design and the deployment of IDSs. We first discussed the benefits of implementing an IDS within both the Cloud and the Fog network paradigm. Later, we identified the main challenges in the design and the deployment of IDS solutions within the Fog computing network architecture. Then, we explored a selected set of existing solutions and we mapped them to the challenges identified. Finally, we discussed the results and proposed some promising future research directions.

Acknowledgement. This publication was partially supported by awards NPRP-S-11-0109-180242, UREP23-065-1-014, and NPRP X-063-1-014 from the QNRF-Qatar National Research Fund, a member of The Qatar Foundation. The information and views set out in this publication are those of the authors and do not necessarily reflect the official opinion of the QNRF.

References

1. Bonomi, F., Milito, R., Zhu, J., Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, pp. 13–16. ACM (2012)
2. Rios, R., Roman, R., Onieva, J.A., Lopez, J.: From SMOG to Fog: a security perspective. In: 2017 Second International Conference on Fog and Mobile Edge Computing (FMEC), pp. 56–61, May 2017
3. Roman, R., Lopez, J., Mambo, M., Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges. Future Gener. Comput. Syst. **78**, 680–698 (2018)
4. Mukherjee, M., et al.: Security and privacy in fog computing: challenges. IEEE Access **5**, 19293–19304 (2017)
5. Munir, K.: Advancing Consumer-Centric Fog Computing Architectures. IGI Global (2018)
6. Sciancalepore, S., Piro, G., Vogli, E., Boggia, G., Grieco, L.A., Cavone, G.: LICITUS: a lightweight and standard compatible framework for securing layer-2 communications in the IoT. Comput. Netw. **108**, 66–77 (2016)
7. Yu, W., et al.: A survey on the edge computing for the internet of things. IEEE Access **6**, 6900–6919 (2018)
8. Hosseinpour, F., Vahdani Amoli, P., Plosila, J., Hämäläinen, T., Tenhunen, H.: An intrusion detection system for fog computing and IoT based logistic systems using a smart data approach. Int. J. Digit. Content Technol. Appl. **10**, 34–46 (2016)
9. Yi, S., Qin, Z., Li, Q.: Security and privacy issues of fog computing: a survey. In: Xu, K., Zhu, H. (eds.) WASA 2015. LNCS, vol. 9204, pp. 685–695. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21837-3_67
10. Martin, A., Raponi, S., Combe, T., Di Pietro, R.: Docker ecosystem-vulnerability analysis. Comput. Commun. **122**, 30–43 (2018)
11. Krontiris, I., Giannetsos, T., Dimitriou, T.: LIDeA: a distributed lightweight intrusion detection architecture for sensor networks. In: Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, p. 20. ACM (2008)
12. Hai, T.H., Huh, E.N., Jo, M.: A lightweight intrusion detection framework for wireless sensor networks. Wirel. Commun. Mob. Comput. **10**(4), 559–572 (2010)
13. Onat, I., Miri, A.: An intrusion detection system for wireless sensor networks. In: IEEE International Conference on Wireless and Mobile Computing, Networking And Communications, WiMob 2005, vol. 3, pp. 253–259. IEEE (2005)
14. Anwar, S., et al.: From intrusion detection to an intrusion response system: fundamentals, requirements, and future directions. Algorithms **10**(2), 39 (2017)
15. Sandhu, R., Sohal, A.S., Sood, S.K.: Identification of malicious edge devices in fog computing environments. Inf. Secur. J.: Glob. Perspect. **26**(5), 213–228 (2017)
16. Sudqi Khater, B., Abdul Wahab, A., Idris, M., Abdulla Hussain, M., Ahmed Ibrahim, A.: A lightweight perceptron-based intrusion detection system for fog computing. Appl. Sci. **9**(1), 178 (2019)

17. Sedjelmaci, H., Senouci, S.M., Al-Bahri, M.: A lightweight anomaly detection technique for low-resource IoT devices: a game-theoretic methodology. In: 2016 IEEE International Conference on Communications (ICC), pp. 1–6. IEEE (2016)
18. Pandeeswari, N., Kumar, G.: Anomaly detection system in cloud environment using fuzzy clustering based ANN. Mob. Netw. Appl. **21**(3), 494–505 (2016)
19. Lo, C.C., Huang, C.C., Ku, J.: A cooperative intrusion detection system framework for cloud computing networks. In: 2010 39th International Conference on Parallel Processing Workshops, pp. 280–284. IEEE (2010)
20. Di Pietro, R., Mancini, L.V.: Intrusion detection systems, vol. 38. Springer, Heidelberg (2008). https://doi.org/10.1007/978-0-387-77265-3
21. Abeshu, A., Chilamkurti, N.: Deep learning: the frontier for distributed attack detection in fog-to-things computing. IEEE Commun. Mag. **56**(2), 169–175 (2018)
22. Kovanen, T., David, G., Hämäläinen, T.: Survey: intrusion detection systems in encrypted traffic. In: Galinina, O., Balandin, S., Koucheryavy, Y. (eds.) NEW2AN/ruSMART -2016. LNCS, vol. 9870, pp. 281–293. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46301-8_23
23. Sciancalepore, S., Oligeri, G., Di Pietro, R.: Strength of crowd (SOC)–defeating a reactive jammer in IoT with decoy messages. Sensors **18**(10), 3492 (2018). Special Issue on Emerging Methodologies and Practical Solutions for M2M and D2D Communications in the Internet of Things Era
24. Baktir, A.C., Ozgovde, A., Ersoy, C.: How can edge computing benefit from software-defined networking: a survey, use cases, and future directions. IEEE Commun. Surv. Tutor. **19**(4), 2359–2391 (2017, Fourthquarter)
25. Caprolu, M., Raponi, S., Di Pietro, R.: Fortress: an efficient and distributed firewall for stateful data plane SDN. Secur. Commun. Netw., 16 (2019)