# Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges

CrossMark

Rodrigo Roman [a,*], Javier Lopez [a], Masahiro Mambo [b]

[a] Computer Science Department, University of Malaga, Ada Byron building, 29071 Malaga, Spain
[b] Faculty of Electrical and Computer Engineering, Institute of Science and Engineering, Kanazawa University, Kakuma Kanazawa 920-1192, Japan

## HIGHLIGHTS

- Features and problems that are common to all edge paradigms are identified.
- Security threats and challenges that affect edge paradigms are analyzed.
- Potential synergies in the development of security mechanisms are shown.
- Issues to be studied and evaluated in the near future are discussed.

## ARTICLE INFO

## ABSTRACT

For various reasons, the cloud computing paradigm is unable to meet certain requirements (e.g. low latency and jitter, context awareness, mobility support) that are crucial for several applications (e.g. vehicular networks, augmented reality). To fulfill these requirements, various paradigms, such as fog computing, mobile edge computing, and mobile cloud computing, have emerged in recent years. While these edge paradigms share several features, most of the existing research is compartmentalized; no synergies have been explored. This is especially true in the field of security, where most analyses focus only on one edge paradigm, while ignoring the others. The main goal of this study is to holistically analyze the security threats, challenges, and mechanisms inherent in all edge paradigms, while highlighting potential synergies and venues of collaboration. In our results, we will show that all edge paradigms should consider the advances in other paradigms.

© 2016 Elsevier B.V. All rights reserved.

## 1. Introduction

Cloud computing has taken the world by storm. In this category of utility computing, a collection of computing resources (e.g. network, servers, storage) are pooled to serve multiple consumers, using a multi-tenant model. These resources are available over a network, and accessed through standard mechanisms [1]. The cloud computing paradigm provides a variety of deployment models and service models, from public clouds (organizations provide cloud computing services to any customer) to private clouds (organizations deploy their own private cloud computing platform), and from Infrastructure as a Service models (IaaS, where fundamental computing resources are offered as a capability) to Software as a

Service models (SaaS, where applications are offered as a capability), among other things. The benefits of cloud computing – minimal management effort, convenience, rapid elasticity, pay per use, ubiquity – have given birth to a multi-billion industry that is growing worldwide [2].

Despite its benefits, cloud computing is not a panacea. Generally, public cloud vendors have built a few large data centers in various parts of the world. These large-scale, commodity-computer data centers have enough computing resources to serve a very large number of users. However, this centralization of resources implies a large average separation between end user devices and their clouds, which in turn increases the average network latency and jitter [3]. Because of this physical distance, cloud services are not able to directly access local contextual information, such as precise user location, local network conditions, or even information about users' mobility behavior. For various delay-sensitive applications, such as vehicular networks and augmented reality, these requirements (low latency and jitter, context awareness, mobility support) are needed.
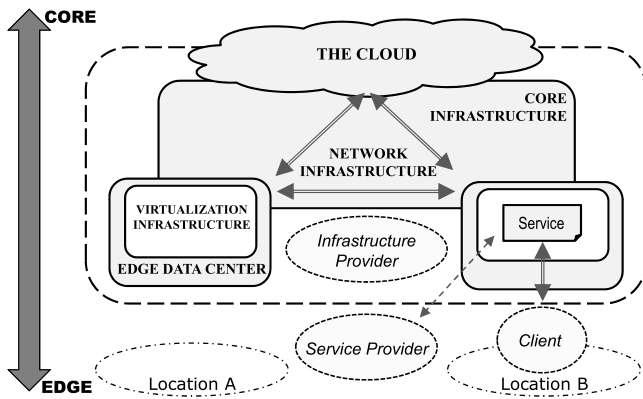
**Fig. 1.** Functional structure of edge paradigms.

For these reasons, in recent years, various novel paradigms have emerged, such as fog computing [4], mobile edge computing [5], and mobile cloud computing [6], among others (cf. [7,8]). The common denominator in these edge paradigms is the deployment of cloud computing-like capabilities at the edge of the network. Most edge paradigms follow the structure shown in Fig. 1. Edge data centers, which are owned and deployed by infrastructure providers, implement a multi-tenant virtualization infrastructure. Any customer – from third-party service providers to end users and the infrastructure providers themselves – can make use of these data centers' services. In addition, while edge data centers can act autonomously and cooperate with one another, they are not disconnected from the traditional cloud. It is therefore possible to create a hierarchical multi-tiered architecture, interconnected by a network infrastructure. Besides, we have to consider the potential existence of an underlying infrastructure, or core infrastructure (e.g. mobile core networks, centralized cloud services), that provide various support mechanisms, such as management platforms and user registration services. Finally, one trust domain (i.e. edge infrastructure that is owned by a infrastructure provider) can cooperate with other trust domains, creating an open ecosystem where multitude of customers can be served.

There are various differences among edge paradigms, such as the focus on mobile network operators as infrastructure providers in mobile edge computing, the existence of user-owned edge data centers (i.e. personal cloudlets) in mobile cloud computing, and the use of different underlying protocols and interfaces, among others. Nonetheless, there remain numerous similarities. Still, little of the research in these fields takes into consideration these similarities. Most architectures, protocols, services, and mechanisms are designed with only one edge paradigm in mind, and they do not consider the state of the art of other edge paradigms. At this initial stage, researchers should consider that research findings in relation to one edge paradigm might also be applied or adapted to other edge paradigms.

This silo mentality is especially conspicuous in the field of security. Although research on security issues in edge paradigms is still nascent, given the importance of this particular field, various researchers have already identified various potential threats. In the process, they have developed several security and privacy mechanisms. However, as mentioned, most research does not follow an interdisciplinary approach: studies tend to focus solely on one particular edge paradigm and its state of the art. Moreover, very few researchers have considered that it might be possible to analyze and adapt other security mechanisms that were initially designed for enabling technologies (e.g. wireless networks, distributed and peer-to-peer systems, virtualization platforms [4]) and other related paradigms (e.g. cloud computing, grid computing).

Therefore, *this study looks to provide, from a holistic perspective, a detailed analysis of the security of edge paradigms*. This analysis will be organized as follows. Section 2 introduces the most important edge paradigms, including their history, use cases, and standardization efforts. Section 3 analyzes the common features of, and differences among, all edge paradigms, and highlights both their challenges and potential synergies. Section 4 introduces the security issues that affect all edge paradigms; this section analyzes the various threat models that target edge paradigms, alongside a brief overview of the requirements and challenges of the security mechanisms that should be used in this context. Section 5 presents an analysis of the current state of the art regarding security in edge paradigms. This analysis does not merely enumerate existing security mechanisms; it also points out synergies among security mechanisms originally designed for edge paradigms and other related fields. Finally, conclusions are presented in Section 6.

*Related work.* In recent years, various authors have surveyed and reviewed the state of the art of the security of various edge paradigms, such as mobile cloud computing [9–11] and fog computing [12–14]. Such works look to provide a preliminary analysis of the threats that affect the integrity of these paradigms, alongside an overview of the security mechanisms by which to protect all actors and infrastructures. Other works focused on specific areas, such as network security [16] and forensics [17] in fog computing. Moreover, certain authors [15] have also provided an brief overview of the basic features of all edge paradigms. However, as shown in Table 1, this is the first study to provide a detailed and up-to-date analysis of several subjects from a holistic perspective, including (i) the common features, differences, and synergies of edge paradigms, (ii) a detailed analysis of the various threat models that target the integrity of all edge paradigms, and (iii) a thorough analysis of the state of the art of security in all edge paradigms, including potential synergies among security mechanisms.

## 2. Overview of edge paradigms

### 2.1. Fog computing

The concept of Fog Computing was introduced by Cisco Systems in 2012, and in its initial definition it was considered as an "extension of the cloud computing paradigm (that) provides computation, storage, and networking services between end devices and traditional cloud servers" [18]. Therefore, fog computing does not cannibalize cloud computing, but complements it: the fog architecture facilitates the creation of a hierarchical infrastructure, where the analysis of local information is performed at the 'ground', and the coordination and global analytics are performed at the 'cloud'. Here, cloud services are deployed mostly at the edge of the network, but they can also be deployed in other locations, such as IP/multiprotocol label switching (MPLS) backbones. In fact, the fog network infrastructure is heterogeneous, where high-speed links and wireless access technologies will coexist [19].

The initial definition of fog computing was later expanded and revised by various researchers (cf. [4,20]). Although this extended definition is debatable, it reveals all the advances that the fog might introduce. Under this new definition, fog computing does not become a mere extension of cloud computing, but a paradigm of its own. The elements that implement the cloud services, the fog nodes, can now range from resource-poor devices (e.g. end devices, local servers) to more powerful cloud servers (e.g. Internet routers, 5G base stations). Also, all these elements can also be able to interact and cooperate with each other in a distributed fashion. This generates a three-tier architecture (Clients ⇔ fog nodes ⇔ Central Servers) where centralized cloud servers coexist with fog nodes but are not essential for the execution of fog services [21]. Moreover, fog computing also provides support for the creation of federated infrastructures, where multiple organizations with their own fog deployments can cooperate with each other.

**Table 1**
Contribution of available surveys on edge security.

| | [9] | [10] | [11] | [12] | [13] | [14] | [15] | Our work |
|---|---|---|---|---|---|---|---|---|
| Features, Synergies | No | No | No | No | No | No | *Partial* | *Yes* |
| Fog—Threats | No | No | No | No | *Partial* | *Partial* | No | *Yes* |
| Fog—Security | No | No | No | *Partial* | No | *2015* | No | *Q3 2016* |
| MEC—Threats | No | No | No | No | No | No | No | *Yes* |
| MEC—Security | No | No | No | No | No | No | No | *Q3 2016* |
| MCC—Threats | *Partial* | *Partial* | *Partial* | No | No | No | No | *Yes* |
| MCC—Security | *2013* | *2013* | *2015* | No | No | No | No | *Q3 2016* |

Originally, fog computing was defined as a platform that enabled the creation of new applications and services in the context of the Internet of Things (IoT). Examples of such services include hierarchical Big Data analytics systems and smart infrastructure management systems (e.g. wind farms, traffic lights) [18, 4]. Yet, at present, there are several studies that examined how this paradigm could be used to implement other types of services: low-latency augmented interfaces for constrained (mobile) devices (e.g. brain–computer interfaces using wireless electroencephalogram headsets [22], augmented reality and real-time video analytics [23]), cyber-physical systems [24], novel content delivery and caching approaches under the context of fog computing [25], and various vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) services such as shared parking systems [26].

As of 2016, the efforts of creating a set of standardized open fog computing frameworks and architectures have started (cf. Open Fog Consortium [27]). These efforts do not need to start from zero, as various researchers have already analyzed what a fog computing architecture could look like. One example is the architecture defined by Sang Chin et al. [28]. This context-aware infrastructure supports a diversity of edge technologies (e.g. Wi-Fi, LTE, ZigBee, Bluetooth Smart), and also supports network virtualization and traffic engineering through Network Function Virtualization (NFV) and Software Defined Networking (SDN) mechanisms. Other researchers have studied how fog computing could be integrated with existing IoT frameworks, such as OpenM2M [29]. In this particular example, fog nodes are deployed at edge devices such as road side units in vehicular networks, and implement various machine-to-machine services such as lightweight M2M device management systems and M2M sensor measurement frameworks.

There are also various researchers that have already identified not only potential challenges, but also forward-thinking deployments that make use of the fog computing paradigm in novel ways. One example is the need to provide an set of APIs that will allow Virtual Machines (VM) to access to services provided by fog nodes. Using these APIs, VMs can access to local information such as network statistics, sensor data, etc [30]. Another example is the deployment of *Airborne Fog Computing* systems – where flying devices such as drones act as fog nodes and collaborate with each other and with other servers in order to provide various services to mobile users [31].

## 2.2. Mobile edge computing

The term *Mobile Edge Computing* (MEC) was firstly used to describe the execution of services at the edge of the network in 2013, when IBM and Nokia Siemens Network introduced a platform that could run applications within a mobile base station [32]. This initial concept only had a local scope, and did not consider other aspects such as application migration, interoperability, and others. MEC acquired its current meaning afterwards, in 2014, when the ETSI launched the Industry Specification Group (ISG) for Mobile-Edge Computing [33]. Under this specification, MEC aims to "provide an IT service environment and cloud-computing capabilities at the edge of the mobile network". This group also pursues the creation

of an open ecosystem, where service providers can deploy their applications across multi-vendor MEC platforms. Once the standard is finished, telecommunication companies will be in charge of deploying this service environment in their infrastructure.

The benefits of deploying cloud services at the edge of mobile networks like 5G include low latency, high bandwidth, and access to radio network information and location awareness. Thanks to this, it will be possible to optimize existing mobile infrastructure services, or even implement novel ones. An example is the Mobile Edge Scheduler [34], which minimizes the mean delay of general traffic flows in the LTE downlink. Moreover, the deployment of services will not be limited to mobile network operators, but it will also be opened to 3rd party service providers as well. Some of the expected applications include augmented reality, intelligent video acceleration, connected cars, and Internet of Things gateways, amongst others [35].

In order to implement the MEC environment, it is necessary to deploy virtualization servers (i.e. MEC servers) at multiple locations at the edge of the mobile network. Some deployment locations considered by the MEC ISG are LTE/5G base stations (eNodeB), 3G Radio Network Controllers (RNC), or multi-Radio Access Technology (3G/LTE/WLAN) cell aggregation sites—which can be located indoors or outdoors. Besides, the MEC ISG has suggested that this virtualization infrastructure should host not only MEC services, but also other related services such as Network Function Virtualization (NFV) and Software Defined Networking (SDN) [35]. Such deployment would reduce the deployment costs, and provide a common management and orchestration infrastructure for all virtualized services.

As of 2016, the ETSI Mobile Edge Computing ISG [33] has produced a MEC framework and reference architecture, whose functional elements provide support to services such as application execution, radio network information, and location awareness. Besides, there are various studies that are investigating how this service environment could be deployed using both existing and novel technologies. For example, Staring et al. [36] evaluated three major open source cloud computing platforms (OpenStack, Eucalyptus and OpenNebula), and identified what modules need to be improved in order to deploy the platforms in a mobile network. Puente et al. [37] analyzed how small cell clouds (clusters of interconnected eNodeB) could be seamlessly integrated in existing LTE-A infrastructures without modifying any existing standards and interfaces. Moreover, Maier and Rimal [38] studied how fiber optic communication technologies could be used to interconnect all the elements of a MEC environment.

## 2.3. Mobile cloud computing

Mobile Cloud Computing (MCC) mainly focuses on the notion of 'mobile delegation': due to the limited resources available to mobile devices, they should delegate the storage of bulk data and the execution of computationally intensive tasks to remote entities. In the original MCC concept, introduced in 2009, only centralized cloud computing platforms were considered as the most viable solution to implement the remote execution of

tasks [39]. Later, other researchers expanded the scope of MCC. In this new vision, tasks could also be delegated to devices located at the edge of the network [40]. At present, both visions of MCC coexist [41]. In this study, we will mostly focus on the latter.

Initially, MCC sought to provide novel solutions to services such as mobile learning, mobile healthcare, searching services, and others [42]. Nowadays, many of these services can be implemented in a centralized cloud (e.g. voice-based search) or in the mobile devices themselves (e.g. text-to-speech engines). Nevertheless, the concept of MCC is still relevant, as its potential has not been fully exploited. There are certain applications, such as augmented reality and augmented interface applications, where the existence of an execution platform located at the vicinity of the mobile devices can provide several benefits such as lower latency and access to context information. Moreover, as mobile devices are equipped with functional units such as sensors and high resolution cameras, it is possible to develop novel crowdsourcing and collective sensing applications that make use of the location information [6].

One of the most active areas of research in the field of MCC is the delegation of tasks to external services [41]. There are various solutions that allow applications to migrate part of their code from the mobile devices to cloud-based computing resources located at the edge. Applications are usually implemented using frameworks like .NET and JVM, which makes the code migration process easier. Some research results allow mobile devices to migrate only part of their code, thus is necessary to statically or dynamically identify the code that needs to be offloaded. Other researchers take a more extreme stance: an entire execution environment (i.e. clone), representative of the mobile device, is created. Then, part of the mobile application (including memory image, CPU state, and others) is loaded into the clone. Finally, some approaches make use of mobile agents infrastructures, where the mobile device create a mobile agent that will acquire/process information on its behalf. There are even approaches, such as the concept of Aqua Computing, that mix the notion of mobile agents and clones [43].

Another important research area is the implementation of the cloud-based computing resources located at the edge. There are two major strategies: proximate immobile computing entities (fixed virtualization servers), and proximate mobile computing entities (ad-hoc conglomerate of mobile devices) [44]. In this article we will focus mostly on the first strategy, but will take into account various aspects of the second strategy.

The core element of the first strategy is the cloudlet. This concept, which was firstly defined by Satyanarayanan et al. in 2009 [45], refers to a small cloud infrastructure located near the mobile users. This small infrastructure can be deployed at business premises (e.g. coffee shops, company buildings), uses persistent caching of data and code instead of hard state, and allows devices to load a small VM overlay over pre-existing full-fledged VM images [3]. There are already proofs-of-concept freely available to the research community [46], including user-centric personal cloudlets. Moreover, several tests have shown that cloudlets improve the response time and the energy consumption of mobile devices (51% and up to 42%, respectively [47]) in comparison to centralized clouds.

There are various instances of the second strategy. They all specify a distributed computing platform on a cluster made of nearly devices, which play the role of servers based on cloud computing principles. The elements of the cluster can be mobile devices (cf. Hyrax [48], FemtoClouds [49]), IoT devices and entities (cf. Aura [50]), or a combination of several types of devices. Due to the limited resources available to the devices that form the distributed cluster, this strategy does not make use of virtualization techniques. Instead, some implementations make use of specific parallel algorithms such as MapReduce, while
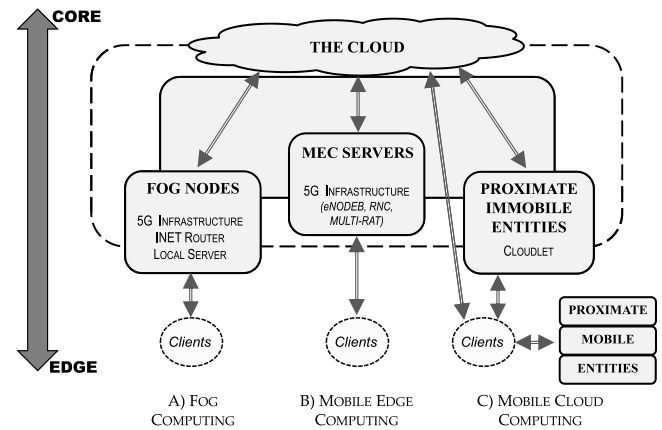


**Fig. 2.** Simplified overview of major edge paradigms.

others take a more general approach and allow various types of computationally intensive tasks. In almost all cases, a controller is in charge of receiving the tasks and discover what devices could optimally execute them.

### 2.4. Other approaches

As we have seen in the previous sections, there are many paradigms that aim to bring cloud services and resources closer to the users. Although we have provided a summary of the most important ones (cf. Fig. 2), there are still some minor nascent architectures that are related to these major paradigms. One example is the concept of the *superfluid cloud*, defined by Manco et al. [7]. In the vision of the superfluid cloud architecture, a set of virtualization platforms with heterogeneous capabilities (from microservers such as Raspberry Pis to larger x86 deployments) are deployed at various points of the network: at the access network level (e.g. 5G eNodeB), at the aggregation network level (e.g. local data centers, network routers), and at the core infrastructure level (e.g. cloud data centers). Besides the deployment of heterogeneous servers, another major differentiator of this architecture is the concept of massive consolidation: the ability to execute a large number (around 10.000) of minimalistic VMs in a single commodity server. These VMs can be deployed and migrated to various points of the network very quickly. This massively distributed, hierarchical architecture enables the creation of on-the-fly services, which might behave as mobile agents if necessary.

Another architecture, known as *edge-centric computing*, was defined by Garcia Lopez et al. [8]. In their vision, a federation of edge-centric distributed services, deployed across data centers and nano data centers, collaborate with each other in a peer-to-peer fashion. Moreover, the cloud can take an auxiliary role, providing stable resources when necessary. This vision enables the creation of human-centered applications, such as the creation of personal spaces at the edge (e.g. personal information with access control and trust mechanisms managed by the users), social spaces at the edge (e.g. crowdsourcing applications based on user-controlled social activities), and public spaces at the edge (e.g. collaborative information flows where multiple actors – human and city services – interact).

## 3. Analysis of features and synergies

### 3.1. Features: similarities and differences

Table 2 summarizes the main properties of every major edge paradigm. Some of these properties were introduced in the previous section, while other properties have been gathered from

**Table 2**
Comparison of features of Edge paradigms.

| | MEC | Fog computing | MCC | Cloud |
|---|---|---|---|---|
| Ownership | Telco companies | Private entities, individuals | | Private entities |
| Deployment | Network edge | Near-edge, edge | Network edge, devices | Network core |
| Hardware | Heterogeneous servers | | Servers, user devices | Servers |
| Service | Virtualization | | Virtualization, others | Virtualization |
| Net. Architecture | N-tier, decentralized, distributed | | | Centralized |
| Mobility | Yes | | | N/A |
| Latency, jitter | Low | | | Average |
| Local awareness | Yes | | | N/A |
| Availability | High | | | |
| Scalability | High | | | Average |

existing reports and research documents (cf. [35,37,27,21,41,42] and others). Note that, for the sake of comparison, this table also includes the properties of the existing centralized cloud computing paradigm.

*Similarities.* When analyzing the properties of the different paradigms, one apparent conclusion is that these paradigms might come from different backgrounds, but they all have the same basic goal: to bring cloud computing-like capabilities to the edge of the network. They all provide support for some type of multi-tenant virtualization infrastructure (e.g. fog node, MEC server, cloudlet), which is easily accessible through various broadband networks (e.g. fiber optic, wireless communications, high-speed mobile networks). These infrastructures can adjust the provisioning of capabilities to the location and needs of their users, accessing nearby computational resources (e.g. neighbor virtualization pools, distributed mobile devices) if needed. Besides, all paradigms take into consideration the need to monitor the use of the different resources, although the entities in charge of this monitoring and the distribution of these entities varies from paradigm to paradigm.

There are various similarities between all paradigms, as well. One clear example is mobility: as most services are provided locally, it is essential to take into consideration the existence of mobile devices. Every paradigm makes use of several strategies to support user mobility: from mobility management entities located at a higher level in the network hierarchy, to mechanisms that provide support for the migration of VMs. Another example is the network architecture. All paradigms can behave as an extension of the cloud, complementing its services, which enables the creation of a hierarchical multi-tiered architecture. On the other hand, the elements of these paradigms can also behave in a decentralized and distributed way; edge data centers can provide services and take decisions autonomously, and also collaborate with each other without completely depending on a central infrastructure. Moreover, all paradigms pursue the creation of federated infrastructures, where multiple edge infrastructures can coexist and exchange information and services.

These paradigms also provide a similar set of benefits, which are derived mostly from the proximity of the edge data centers. For example, whenever users access to the computing capabilities of their surroundings, both the network latency and the packet delay variation (jitter) are low and predictable. Another benefit is the ability to access to local information (e.g. network conditions, physical aspects of the environment, geographical location), which allows all users and services to be aware of their local context. A further benefit is the scalability of the whole ecosystem. There are various reasons for this. First, nodes can be wide-spread and geographically available in large numbers. Second, it is assumed that the nodes located at a certain site will mostly provide services to local devices. Note that it can be possible to make use of neighboring nodes, or even use nodes situated at more remote geographical locations or at a higher level in the hierarchy, if the situation requires it. Finally, another important benefit is the high

availability of the services. There are two reasons for this: (i) node redundancy at a local level, and (ii) in certain paradigms (e.g. MEC), the edge data centers are actually hosted by the communication infrastructure (e.g. mobile network infrastructure).

*Differences.* Obviously, even if all these paradigms have the same goal, they will have some underlying differences on how they want to fulfill that goal. For example, MEC limits the deployment of edge computing platforms to mobile network infrastructures such as 5G. On the other hand, fog nodes can also be deployed at other locations, such as user-managed servers, access points, routers, gateways, etc. As for MCC, it has an even more distributed scope, where in some instances the devices themselves can participate in the service provisioning process. This difference on the deployment and management of the edge data centers influences over who can become a service provider. For example, in MEC, only telecommunication operators can become MEC providers, as they own the mobile network infrastructure where the edge data centers are deployed. In contrast, any user (from companies to tech-savvy end users) can deploy their own fog and MCC nodes, effectively becoming part of the service provisioning ecosystem— or even creating their own private cloud-like environments.

Another difference, related to the previous point, is the deployment of curated applications. As MEC servers are controlled by telecommunication operators and hosted in their infrastructure, it is possible for third-party service providers to work closely with the operators and develop MEC-specific services. Such services can then be extensively tested and possibly integrated in a customized way. This is also true for fog computing, as certain fog nodes can be deployed in ISP infrastructures (e.g. routers and gateways). Finally, some paradigms, such as MCC, provide some specific services that are not considered by other paradigms. For example, MCC provides support for distributed execution mechanisms that are not related to virtualization, such as the execution of the MapReduce parallel algorithm over constrained devices. Another example is the edge-centric computing vision, which focus on personal spaces (e.g. user-controlled personal networks) and peer-to-peer interactions.

### 3.2. Challenges and synergies

Due to the similarities between all edge paradigms, they have several major challenges in common, which are summarized in Table 3 (cf. [20,6,35] and others). The common denominator of most of these challenges is the decentralized and distributed nature of these paradigms, in contrast with the centralized nature of the cloud computing paradigm. The decentralization and proximity of the service infrastructure to the edge brings various benefits (e.g. low latency, scalability), but it also brings new issues that must be carefully considered. Examples of such issues are the mobility of the various entities (including the service infrastructure itself [31]) and the need to synchronize 'soft' and 'hard' states

**Table 3**
Common challenges in edge paradigms.

| Challenge | Description |
| --- | --- |
| Infrastructure | Interoperability; Monitoring; Accountability |
| Virtualization | VM lifecycle; Container and context awareness |
| Resources & tasks | Resource location; Task scheduling; Offloading |
| Distribution | Cooperation; N-tier management; 'Soft state' |
| Mobility | Connectivity; Seamless handoff |
| Programmability | Usability; Session management |

within a multi-tiered architecture. The distributed nature of the service infrastructure, where several edge data centers that might be owned by different infrastructure providers should be able to collaborate with each other, imposes other challenges as well. It is necessary to develop standards that specify how the different elements of the architecture can collaborate with each other, and also how the VMs can access certain information (e.g. context and host information) regardless of their deployment place. Precisely, regarding virtualization, it is essential to provide support for an optimized VM lifecycle—where their creation, deployment, and migration is as lightweight as possible. Moreover, as resources are distributed over various entities and locations, there must exist a set of mechanisms that enable the discovery and orchestration of such resources, including their monitorization.

While all paradigms share these common challenges, it still is necessary to consider the nuances of every paradigm (e.g. the features of their underlying protocols, their specific use cases) when researching and developing novel solutions. In fact, due to technical, economic and political reasons, it is clear that multiple standards and specifications will coexist, with their own solutions to existing problems. Yet, even if there are several structural differences between the paradigms, it does not mean that they should exist in a vacuum, ignoring the advances in other related fields. Due to the similarities between the paradigms (cf. Section 3.1), it is safe to assume that there will be mechanisms and platforms that can provide a generic solution to a shared problem. Such solutions can then be adapted to other edge paradigms. In fact, this assumption is supported by the existing state of the art, where there are already various mechanisms that have been developed with a certain paradigm in mind, but that can be applied to other paradigms as well. We will provide various instances of such synergies in the following paragraphs.

One area where clear synergies between paradigms can be found is the management of VMs. There are already various works, in the areas of fog computing [51] and MEC [52], that define and solve optimization problems whose goal is to improve the distribution of VMs over a set of local clusters, minimizing the utilization of resources. These mechanisms also define when VMs need to replicate, migrate and be merged. As the underlying assumptions made by these algorithms mainly demand a decentralized infrastructure able to access local information, they can be adapted to any edge paradigm. Another important aspect is the cost of setting up and deploying new VMs. Precisely, the advances in areas such as the superfluid cloud [7], where thousands of minimalistic VMs can be deployed in a commodity server with minimal latency, can be adapted and used in other paradigms as well.

Another area of research whose solutions might be applied to other paradigms is resource offloading, which is one of the most studied areas in the field of MCC. As mentioned in Section 2.3, there are various strategies that allow user devices to delegate their tasks to external servers (remote procedure call frameworks, code migration, clone deployment, mobile agents). Yet there are situations where not an user device, but a VM might want to delegate part of its tasks. Instances of such situations are user mobility (e.g. specific tasks – not the whole VM – are migrated to the closest node to the user), task optimization (e.g. network intensive tasks are kept at a local level, and computationally intensive tasks are sent to a more capable cloud system [53]), user empowerment (e.g. the creation of user spaces in the network through clones in aqua computing [43]), and others. Many delegation strategies only require the existence of a virtualization platform where tasks can be delegated [6], thus a lot of research on this particular area can be adapted to satisfy the needs of applications deployed in other edge paradigms.

Other examples include user mobility, context awareness and location of resources. There are various algorithms, such as [54,55], that try to predict the location of (potential) users in order to deploy expected resources in advance. The core algorithms that implement the migration plan only require a distributed architecture of computing platforms that are able to communicate with each other with as less latency as possible. As for context awareness, several works in this area analyze how local hardware awareness can help VMs to understand the limits of their own containers [30], or how VMs can make use of hardware acceleration technologies such as graphic computing units [56]. These works are essential in a context where virtualization servers are heterogeneous, and VMs might need to dynamically adjust their behavior. About the location of resources, there have been various works in the area of pervasive computing (cf. [57]) that could be applied to fog, MEC, and other paradigms. Many of these search engines make use of a N-tiered hierarchy, where the lowest layer of the hierarchy stores information about the local contexts. Algorithms such as Bloom filters are then used to represent a set of keywords in order to reduce the communication overhead.

There are also several application scenarios that have been defined for only one paradigm, but due to their requirements (e.g. support for decentralized and distributed execution platforms) they could be implemented in other related paradigms as well. Scenarios that have been defined mostly for fog computing, such as IoT node pairing services [58], context-aware data analytics platforms [59], and emergency notification mechanisms [60] could also be deployed in MEC and cloudlet infrastructures. Lastly, there are various supporting services that, even if they have been developed with a single paradigm on mind, they can be adapted to other paradigms. The network store concept, developed by Nikaein et al. [61] is one of such supporting services. This research work introduces a digital distribution platform for MEC, which provides Virtualized Network Functions (VNFs), or slices, that enable 5G application use-cases. Although these slices cannot be directly deployed in other related paradigms due to the differences in the underlying protocols, the concept of a digital distribution platform and its architectural elements (service and business layer, slice orchestrator and manager layer) can be adapted and deployed in other virtualization platforms. Precisely, due to the distributed and collaborative nature of edge paradigms, the implementation of a network store can serve as a catalyst for the rapid deployment of edge applications, as it can behave as a repository of functionality and knowledge.

As a final note, we want to emphasize that, even if every paradigm pursues the creation of their own standards and their own service infrastructure, this does not mean that it is not possible for them to collaborate with each other—or even being integrated with each other. For example, cloudlets are traditionally associated to MCC, but they can become a technology enabler for both fog computing and MEC. Also, as both fog computing and MEC aim to provide support for federated services and interactions with different providers (e.g. through a set of open APIs), it can be possible to create applications that make use of both edge paradigms, or even deploy middleware platforms that will connect various edge paradigms at an infrastructure level.

## 4. Security threats

There are several challenges that must be overcome in order to create an ecosystem where all actors (end users, service providers, infrastructure providers) benefit from the services provided by edge paradigms. Not surprisingly, one of the greatest challenges is security. In this section, we will (a) review why security is a very important factor in this particular context (Section 4.1), (b) analyze the specific threats that can target edge paradigms (Section 4.2), and (c) introduce the requirements and challenges of the security mechanisms that should be applied to this particular context (Section 4.3).

### 4.1. The importance of security in edge paradigms

As aforementioned, one of the greatest challenges for the creation of a edge paradigm ecosystem is security. There are several reasons for this. First, at the core of most edge paradigms, there are several enabling technologies such as wireless networks, distributed and peer-to-peer systems, and virtualization platforms [4]. It is then necessary not only to protect all these building blocks, but also to orchestrate the diverse security mechanisms. This is by itself a complex issue, as we need to create an unified and transversal view of all the security mechanisms that allows their integration and interoperability.

Second, the whole is greater than the sum of its parts: by assuring the security of all the enabling technologies, we do not assure the security of the whole system. Once cloud computing-like capabilities are brought to the edge of the network, novel situations arise (e.g. collaboration between heterogeneous edge data centers, migrating services at a local and global scale) whose security has not been widely studied. Besides, we also need to consider the specific requirements of this particular context (cf. Section 3.1), which might affect the kind of security mechanisms that could be deployed. For example, the security mechanisms should be as autonomous as possible and not depend on the continuous existence of a centralized infrastructure. There are two main reasons for this: not only there will be situations (e.g. malicious attacks, intermittent connectivity, distributed applications) where no centralized control system is available, but it is also necessary to take into account the latency of the security mechanisms. Another example is the technological limitations of the elements of the infrastructure. For example, certain edge data centers might be composed of microservers (e.g. Raspberry Pi) that lack the hardware protection mechanisms of commodity servers [7], or include legacy edge devices with limited connectivity—which restricts the authentication protocols that can be deployed [12]. Moreover, the security mechanisms need to consider the existence of mobile devices, which can make use of the edge data centers anytime and anywhere.

Third, we need to keep in mind that, in addition to the security threats that will appear due to the specific features of edge paradigms, the whole system also inherits the security threats that are present in their building blocks and in the application scenarios. And this is no trivial issue, because these threats are, in fact, very significant. A clear example of this is the Internet of Things, which is the main *raison d'etre* of fog computing [18]—and a major use case in all edge paradigms. It is also considered as a combination of "the worst-of-all-worlds" in terms of security: not only we need to combine and protect multiple layers of technologies (from network to mobile to cloud [62]), but also provide global connectivity and accessibility in a heterogeneous ecosystem [63]. This situation generates a considerable attack surface, that in turn also affects all paradigms that make use of the IoT.

Finally, the impact than a successful attack might cause in our society is quite considerable. The number of application scenarios where edge paradigms can be applied is huge. In fact, almost any aspect of our daily lives can be influenced by applications deployed in these infrastructures: Our private information (e.g. photos, medical reports), our daily routines (e.g. transportation, shopping), our enterprise ecosystems (e.g. industries, supply chains), our critical infrastructures (e.g. energy, emergency systems), etc. Without proper security and privacy mechanisms, the benefits of edge paradigms will be quickly overshadowed by the damage caused by malicious adversaries.

### 4.2. Threat landscape

Once we have understood the importance of security in the context of edge paradigms, it is time to analyze what are the specific threats that can target these paradigms, and the extend of the damage they can cause. In the near future, this analysis will help us in the development of security mechanisms that can adequately protect the whole ecosystem against such threats. Besides, it will also allow us to understand what are the particularities of every edge paradigm, as they have subtle differences that will affect the implementation and deployment of the security mechanisms.

However, before analyzing the threats, it is necessary to examine how the lack of a global perimeter affects the security of edge paradigms. As we have seen in previous sections, even in the most closed paradigm – mobile edge computing – the whole ecosystem will not be controlled by one single owner. Even more, edge data centers are capable of providing services without continuously depending on a central infrastructure. Therefore, all relevant assets, including the network infrastructure, the service infrastructure (e.g. edge data centers, core infrastructure), the virtualization infrastructure, and the user devices, are controlled not by a single entity but by various actors (including, in some cases, end users) who need to cooperate with each other. A consequence of this situation is that every element of the infrastructure can be targeted or subverted at any moment. In fact, this "anything, anytime" principle is also inherited from some of the underlying building blocks and application scenarios, such as the Internet of Things [63].

Having said that, the "anywhere" principle (attacks can be performed from anywhere) does not fully apply to this particular context. The cause of this is the geographical location of the edge data centers. One of the basic tenets of these paradigms is that cloud computing capabilities are basically provided in close proximity to end users. As a result, a edge data center (e.g. fog nodes, MEC servers) will provide services mostly to local entities (e.g. mobile users located at the vicinity, entities inside a building). There are a few exceptions to this rule, such as virtual machines that act like agents and migrate to other infrastructures away from their physical counterpart, or specific local services that are requested by remote entities. This particularity of edge paradigms is a double-edged sword: on the one hand, it limits the impact of an attack to the local environment. On the other hand, if one adversary can control one edge data center, it might be able to control almost all the services that are provided in that geographical location.

There is another consequence of the lack of a global perimeter: the nature of the different attacker profiles that will target edge paradigms. Even if traditional 'external' attackers will exist (i.e. adversaries that do not control any element of the whole infrastructure), there will exist many adversaries that will control one or more elements of the infrastructure: user devices, virtual machines, servers, sections of the network, even whole edge data centers. This situation is similar to the current Internet, where malicious adversaries can take control of existing elements or deploy their own. These adversaries are both 'internal' and 'external', as they control one part of the infrastructure but not the others. Note that these attackers can still try to influence other healthy sections of the infrastructure. Examples are the injection of bogus information during a collaboration process, or the

deployment of malicious virtual machines that, like viruses, will try to exploit vulnerabilities in their hosts. Needless to say, traditional 'internal' attackers (i.e. undercover adversaries such as disgruntled employees that are officially allowed to access certain elements of part of an infrastructure) will also exist within this ecosystem.

### 4.2.1. Threat model

After reviewing the nature and scope of the potential attackers, we can finally provide an analysis of the threats. For this analysis, we will enumerate the most important assets of edge paradigms, and then summarize the attacks that can be launched against such assets. Note that some of the threats that affect edge paradigms will be the same threats that affect traditional data centers, as both of them share various assets (e.g. server farms, networking infrastructure). Still, in our analyses we need to consider the specific decentralized and distributed nature of edge paradigms, plus the existence of additional services such as interoperability and mobility support, location awareness, and others. Therefore, not only the impact of certain common threats will be different (e.g. an attack to an edge data center will mostly impact the services related to that geographical area), but also novel threats will arise.

A summary of this threat classification can be found in Table 4. Note that this classification will be defined in a way that it can be applied to all edge paradigms—we will explicitly explain the particularities of every major paradigm afterwards, in Section 4.2.2.

*Network infrastructure.* As aforementioned, edge paradigms make use of various communication networks to interconnect their elements: from wireless networks to mobile core networks and the Internet. An adversary can try to target any of these communication infrastructures.

- *Denial of Service (DoS).* All communication networks are vulnerable to several DoS attacks, such as distributed denial-of-service (DDoS) attacks and wireless jamming. Yet the scope of these attacks is limited. Attacks against the edge networks will only disrupt the vicinity of the affected networks. Also, an attack to the core infrastructure might not completely disrupt the functionality of the edge data centers, as their protocols and services can be designed to work in an autonomous or semi-autonomous way.
- *Man in the Middle.* Malicious adversaries can be able to take control of a section of the network, and then launch attacks such as eavesdropping and/or traffic injection. The practicality of this particular threat was demonstrated by Stojmenovic et al. [12]. In this particular case, a gateway that interconnected two 3G and WLAN networks was compromised, and the adversary gained access to the network interfaces. This attack is not only very stealthy but also very dangerous, as it can affect all the elements (e.g. information, virtual machines) that traverse that particular node.
- *Rogue Gateway.* The open nature of several edge paradigms, where even user-owned devices can become full-fledged participants (e.g. personal cloudlets, mobile devices participating in a cluster of nearby devices), create a scenario where malicious adversaries can deploy their own gateway devices. This particular threat produces the same outcome as the Man-in-the-Middle attack (e.g. the ability to eavesdrop and/or inject traffic), even if the means are different (compromising versus deploying).

*Service infrastructure: edge data center.* The edge data center hosts the virtualization servers and several management services, amongst others. However, for an external adversary, the attack surface of a edge data center is quite considerable: from multiple public APIs that provide services to all actors (e.g. users, virtual machines, other data centers) to other access points such as web applications. Note that the specific threats related to the virtualization infrastructure will be described later.

- *Physical damage.* In certain paradigms, the elements of the service infrastructure might not be guarded or protected against physical damage. Clear examples are fog nodes managed by small businesses and user devices forming clusters. For this particular threat, it is necessary for the attacker to be in the vicinity of the device in order to destroy it. As a result, there is a very high probability that this kind of attack will be witnessed by various observers. Moreover, the impact of this particular attack is limited to a local scope: only the services associated to a particular geographical location will be disabled.
- *Privacy leakage.* Both internal adversaries and honest but curious adversaries can try to access the flow of information that traverse the edge data center. Nevertheless, the scope of these attacks is limited: An edge data center mainly stores and processes information from the entities that are located at its vicinity, although in some cases (e.g. distributed storage services, migrating virtual machines) it can deal with data coming from other locations. Note, however, that these edge data centers might be able to extract more sensitive information about a user thanks to their awareness of the context [14].
- *Privilege escalation.* The considerable attack surface of these edge data centers allows external adversaries to try to take control of various of its services. This is facilitated by the fact that edge data centers can be managed by professionals with limited security training, or even hobbyists. These infrastructures might be misconfigured, or even lack proper maintenance. Note that this attack can also be performed by internal adversaries that abuse of their privileges and take advantage of their insider knowledge.
- *Service manipulation.* Once an adversary has gained control of certain sections of the edge data center, either by privilege escalation or by abusing his own privilege as a legitimate administrator, it can manipulate the services of the data center. As a result, the adversary can launch several types of attacks, such as selective denial of service attacks and selective information tampering, amongst others.
- *Rogue data center.* In this threat, an adversary is able to control an entire edge data center through various means, such as privilege escalation or deploying his own malicious infrastructure. This creates a very dangerous scenario, as the adversary (i) has complete control of all the services that are provided in a geographical location, (ii) has access to all information flows that are directed to the rogue data center, and (iii) can manipulate all interactions with external systems (e.g. migrating virtual machines, service requests from remote entities).

*Service infrastructure: core infrastructures.* All edge paradigms can be supported by several core infrastructures, such as mobile core management systems and centralized cloud services. It is then necessary to analyze what are the specific threats that target these upper layers in this particular context. It should be noted that, in certain paradigms (e.g. MEC), the core infrastructure will be managed by the same companies (e.g. mobile network operators) that deploy the edge data centers. Besides, we should not assume that all interactions with a cloud provider can be completely trusted, due to cyber-crimes [64] and other reasons (e.g. government intrusion [65]). A complete taxonomy of general threats that target cloud providers is available elsewhere [66].

- *Privacy leakage.* There are no guarantees that all information flows that are processed and stored in the upper layers of our edge infrastructures will not be accessed by unauthorized entities or honest but curious adversaries. Note, however, that these internal adversaries might not have access to the whole information set, including raw measurements. The reason is simple: as the lower layers, the edge data centers, will process

**Table 4**
Categorization of threats in edge paradigms.

| Asset | Threats |
| --- | --- |
| Network infrastructure | Denial of service, man-in-the-middle, rogue gateway |
| Edge data center | Physical damage, privacy leakage, privilege escalation, service manipulation, rogue data center |
| Core infrastructures | Privacy leakage, service manipulation, rogue infrastructure |
| Virtualization infrastructure | Denial of service, misuse of resources, privacy leakage, privilege escalation, VM manipulation |
| User devices | Injection of information, service manipulation |

the local information, it is probable that the upper layers will only receive a subset of said information. In addition, edge paradigms allow edge data centers to exchange information directly with each other, bypassing the central systems.

- *Service manipulation.* An internal adversary with enough privileges can try not only to manipulate the information flow, but also to instantiate rogue services that will provide bogus information (e.g. fake management information, historic data) to other partners. But this particular threat follows the same principle as the privacy leakage threat: these internal adversaries will not be able to influence the whole ecosystem, due to the decentralized and distributed nature of edge paradigms.
- *Rogue infrastructure.* This threat assumes that certain elements of the core infrastructure can be targeted by specialized adversaries. Such attacks will be able to take control of some services of the upper layers of the infrastructure, causing havock on the whole ecosystem. Although the chances of an adversary successfully launching this attack are extremely low, it is still necessary to have this scenario in mind for especially sensitive situations, where specialized security and fault tolerance mechanisms need to be deployed.

*Virtualization infrastructure.* Within the core of all edge data centers, we can find a virtualization infrastructure, which enables the deployment of cloud services at the network edge. Like all other assets, this infrastructure can be exploited in several ways. Besides, we also need to consider that the virtual machines themselves might be controlled by malicious adversaries who are trying to misuse or exploit the resources available to them.

- *Denial of Service (DoS).* A malicious virtual machine can try to deplete the resources (including computational, network and storage resources) of the host where it is running. This threat is quite significant for this particular context, as most edge data centers will not have the resources that are available to other cloud infrastructures.
- *Misuse of resources.* A malicious virtual machine can execute various malicious programs that do not target the edge data center where it is hosted, but other local or remote entities. For example, a malicious virtual machine can search for vulnerable IoT devices in the local environment. It can also execute programs for cracking passwords, or host botnet servers.
- *Privacy leakage.* Due to requirements in their design, most virtualization infrastructures located at edge data centers are not completely transparent: they can actually implement various APIs that provide information about the physical and logical environment, such as the state of the local network. However, if these APIs are not protected, a malicious virtual machine can be able to obtain sensitive information about the execution environment and the surroundings of the edge data center.
- *Privilege escalation.* Malicious virtual machines can also try to take advantage of vulnerabilities in their hosts. There are various outcomes of this attack: from isolation failures, where the malicious VM succeeds at manipulating other VMs, to escalation of privileges, where the malicious VM takes control of certain elements of the host. This problem is exacerbated by the fact that virtual machines can migrate from one data center to the other due to various reasons (e.g. users moving from one location to the other, virtual machines acting as agents).

- *VM manipulation.* A host system that is being controlled by an adversary (e.g. a malicious insider with enough privileges, a VM that has escalated privileges), can launch several attacks to the VMs that are running inside it. These attacks can range from the extraction of information to the manipulation of the computational tasks are being executed within the VM. Moreover, the adversary can also infect the VM with logic bombs, malware or other malicious elements that will compromise the security of other data centers once the VM migrates to other physical locations.

*User devices.* The devices controlled by the users are also important elements of the whole ecosystem. They not only consume services, but also can become active participants that provide data and participate in the distributed infrastructure at various levels. However, there will be also rogue users that might try to disrupt the services in one way or another. Note, however, that the scope of these threats is quite limited: in this context, users can only influence their immediate surroundings.

- *Injection of information.* Any device that is controlled by an adversary can be reprogrammed to distribute fake information when queried (e.g. vehicles reporting wrong values, users providing fake data to crowdsourcing services). Note that a device might also provide bogus values due to an anomaly in their sensors or internal systems.
- *Service manipulation.* There are some cases where a device might participate in the provisioning of services. For example, a cluster of devices controlled by a virtual machine located at an edge data center can act as a distributed computing platform. Yet if an adversary gains control of one of these devices, it can be able to manipulate the outcome of the service.

### 4.2.2. Differences between paradigms

In this section, we will make use of the features defined in Section 3 to analyze how the threats presented in the previous section affect all paradigms. One feature that has a noticeable effect on the impact of the previous threats is the **ownership** of the infrastructure. In some paradigms, such as *mobile edge computing*, one single company (the mobile network operator) controls not only various edge data centers located at different geographical locations, but also part of the core networks that are connected to those data centers (i.e. the mobile network infrastructure). In principle, this infrastructure is well maintained, with a consistent security policy and guarded against physical and virtual intruders. Thanks to this, the attack surface should be smaller, which decreases the chances of an adversary destroying or gaining control of part of the service infrastructure. Other paradigms, such as *fog computing* and *mobile cloud computing*, allow small companies (e.g. stores) to deploy their own edge data centers, or even allow users to become active participants in the provisioning of services. This creates a more heterogeneous ecosystem, which will probably be less protected than the infrastructures deployed by big companies due to various reasons (e.g. deficient maintenance, limited physical protection).

However, having a large segment of the service infrastructure managed by one single company has its drawbacks, too. One clear example is the impact of a successful attack. Once an adversary

has taken control of a section of the infrastructure, he becomes an internal attacker within that infrastructure. If the necessary contingency mechanisms are not in place, he might try to gain more privileges and/or exploit further vulnerabilities in order to gain even more influence. Moreover, if an insider adversary takes control of certain elements of the core network of that company, it can be able to manipulate large sections of the whole ecosystem. On the other hand, if an adversary takes control of an edge data center managed by a small company or a tech-savvy individual, his reach will be limited to the scope of that particular edge data center.

Another feature that has some influence on the security threats is the **hardware** used to implement the cloud services in the edge data centers. Paradigms such as *mobile cloud computing* and concepts such as the *Superfluid Cloud* can make use of microservers (Raspberry Pi) and user devices (mobile phones) to provide their services. At present, it is still necessary to analyze how the hardware extensions of certain microcontrollers can be used to guarantee a secure virtualization environment [67]. Regarding the hardware used in paradigms such as *fog computing*, the commodity servers used in small-scale deployments can make use of the same security mechanisms as the commodity servers used in cloud deployments [68]. Note, however, that some small-scale deployments might lack experienced staff, and as a result there will be some processes (e.g. definition of security policies, separation of roles, storage of logs in separate physical storage) that might not be properly implemented or maintained.

Regarding the **deployment** of the elements of the infrastructure, we have already mentioned that certain instances of the *mobile cloud computing* paradigm allows the creation of clusters of devices at the very edge of the network, and that these clusters provide services through mechanisms such as parallelization. Because of this, the MCC paradigm has his own extra set of security challenges [10], such as the impact of malware in the user devices, the identification and authentication of the different peers, and the existence of DoS attacks that target honest participants. Besides, we need to mention one aspect that is strongly linked to the **network architecture**. All paradigms support the creation of a hierarchical multi-tiered architecture, where different elements (user devices, edge data centers, core infrastructures) have different roles. As such, certain security services (e.g. authentication, monitorization) can be deployed in a more centralized or a more distributed way. Every approach has its own advantages and disadvantages. For example, if a centralized service is rendered unavailable or is controlled by an adversary, the whole infrastructure will collapse unless contingency mechanisms are in place. Finally, we also need to consider that certain paradigms will make use of their own **protocols** and **services**, such as the Small Cell as a Service (SCaaS) elements in MEC environments, which will have their own security requirements (cf. [69]).

## 4.3. Security mechanisms

In order to create an effective layer of defense against the different threats, it is crucial to deploy various types of security services and mechanisms. In this section, we will introduce the security services and mechanisms that should be integrated in all edge paradigms, alongside with a brief overview of their requirements and challenges in this particular context. Note that all security mechanisms need to take into account various common requirements and constraints, such as reducing the latency of their operations as much as possible, supporting mobile devices and other mobile entities (e.g. virtual machines), achieving technical, functional, and semantic interoperability, managing the limitations of existing technologies, and providing support for disconnected operations.

*Identity and authentication*. In all edge paradigms, there are multiple actors (end users, service providers, infrastructure providers), services (virtual machines, containers), and infrastructures (user devices, edge data centers, core infrastructures) interacting in an ecosystem where multiple trust domains coexist. This situation brings numerous challenges, as not only we need to assign an identity to every entity, but also we need to allow all entities to mutually authenticate each other. Without these security mechanisms, it would be very easy for external adversaries to target the resources of the service infrastructure with impunity. Moreover, internal adversaries would not leave a trail of evidence behind their malicious acts.

In this context, it is necessary to explore identity federation mechanisms and inter-realm authentication systems, which should be interoperable with each other. Besides, due to various requirements (latency, availability of a central server), it is also desirable that an entity can provide a proof of its identity without contacting a central server (e.g. presenting valid and trusted attributes). Note, however, that in some cases parts of the infrastructure can be managed by end-users (e.g. personal cloudlets), and even interact in a peer-to-peer fashion. Therefore, we should study the applicability of distributed authentication mechanisms.

*Access control systems*. The existence of an authorization infrastructure is equally important for edge paradigms, as it is essential to check the credentials of the various entities in order to authorize their requests to perform certain actions (e.g. service providers deploying virtual machines, virtual machines accessing edge data center APIs, edge data centers interacting with each other). If there are no authorization mechanisms in place, anyone without proper credentials can misuse the resources of the virtualization infrastructure. Users would be able to impersonate administrators and control the services of the infrastructure. Malicious attackers would be able to access any resources, including proprietary and/or personal information. The possibilities would be limitless.

Due to the inherent features of edge paradigms, it is crucial to deploy an authorization infrastructure in every trust domain, so as to allow the owners of such domains to disseminate, store and enforce their own security policies. Such infrastructures should be able, in principle, to process the credentials of any entity if there is a trust relationship between them. Moreover, it should be also possible to take into account various factors, such as the geographical location and the resource ownership, in the definition of the authentication policies. For example, migrating virtual machines might be allowed to use additional resources from the virtualization infrastructure if they hold certain privileges (e.g. owned by local law enforcement agencies).

*Protocol and network security*. If the network infrastructure is not protected, the whole service ecosystem will be threatened by internal and external malicious adversaries. It is then necessary to protect the myriad of communication technologies and protocols that are used by edge paradigms. For example, there are various wireless communication technologies (e.g. Wi-Fi, 802.15.4, 5G, Sigfox, LoRa) that might be used to serve local customers. Therefore, edge data centers and their administrators need to understand and make use of the security protocols and extensions implemented by such technologies. Also, edge paradigms need to configure and integrate the security protocols that are used by the core infrastructures (e.g. public Internet, mobile network infrastructure). Moreover, we need to provide network isolation among tenants in the virtualization infrastructure, among other protection mechanisms.

Here, there are various challenges that need to be addressed. For starters, it is necessary to adequately configure the different elements of the network infrastructure. Yet all these elements

will be deployed in different geographical locations, which will be managed by different administrators. Besides, there will situations where entities that belong to different trust domains (e.g. edge data centers from different infrastructure owners) will interact with each other. In this very heterogeneous scenario, we need to establish a secure connection between entities that might even use different communication technologies. There are other aspects that are just as important, such as achieving a dynamic balance between the strength of the security mechanisms and the overall quality of service of the network.

*Trust management.* Another security mechanism that is of great importance for edge paradigms is trust. In this context, the concept of trust goes beyond the idea of "not knowing who I am interacting with", which is mostly solved by implementing authentication mechanisms and establishing trust relationships between trust domains. The reason is simple: we also have to deal with the concept of uncertainty, or "not knowing how my partner is going to behave". All entities have a variety of collaborating peers at their disposal: users can have various service providers available in their vicinity, service providers can choose from many infrastructure providers, and so on. However, such peers might not meet our expectations: the service latency might be high, the anomaly detection rate might be low, or the data might be inaccurate. There are even worse situations: peers might behave egoistically or maliciously.

It is then necessary to seriously consider the deployment of trust management infrastructures in this context. The benefits are numerous: from improving the decision-making processes of all entities (e.g. migrate high priority virtual machines to nearer edge data centers with higher reputation), to enhancing the management of personal data (e.g. reduce the granularity of the information that is transmitted to low reputation entities), amongst others. There are many challenges, though. All trust management infrastructures should be able to exchange compatible trust information with each other, even if located at different trust domains. Another problem lies with the storage and dissemination of trust information, as it should be accessible anywhere, anytime, with as less latency as possible. Moreover, due to the dynamic nature of the infrastructures, non-malicious entities might find themselves with a low reputation due to temporary reasons, thus it is necessary to find a balance between punishment and redemption.

*Intrusion detection systems.* Talking about malicious entities, we have already seen in Section 4.2.1 that external and internal adversaries can attack any entity at any time. Without proper intrusion detection and prevention mechanisms, any successful attacks will go undetected, slowly undermining the functionality of the whole infrastructure. It is then necessary to ensure that the whole infrastructure is covered by such defense mechanisms. Fortunately, we also have seen that the "anywhere" principle does not completely apply to these paradigms: the impact of most attacks is usually limited to a local environment. Therefore, local infrastructures, such as edge data centers, can be in charge of monitoring all their elements – network connections, virtual machines, etc. – and their surroundings. Besides, these local infrastructures can also cooperate with each other or with core infrastructures located at a higher level in the network hierarchy. This way, it can be possible to detect attacks that target large sections of the service infrastructure.

However, the challenges of running an interconnected network of detection and prevention mechanisms in a heterogeneous, decentralized and distributed infrastructure are numerous. The specific attacks that can be launched against edge paradigms need to be understood. If a database of attacks is used (e.g. for signature-based IDS), it needs to be updated and protected at all times. A balance between local and global defense mechanisms needs to be achieved, and a global monitoring infrastructure that encompasses multiple layers and/or trust domains needs to be developed. Moreover, all defense mechanisms, regardless of their location, must be able to exchange information with each other in an interoperable format. Such information should be permanently available in order to detect more persistent threats. Finally, the defense mechanisms must behave as autonomously as possible, in order to reduce the maintenance overhead and improve the usability of the security infrastructure.

*Privacy.* Besides malicious adversaries, it is also possible to find honest but curious adversaries. These adversaries are usually authorized entities (e.g. edge data centers, infrastructure providers) whose secondary goal is to know more about the entities that make use of their services. This knowledge can then be used in various ways: usage profiling, location tracking, disclosure of sensitive information, etc. All these adversaries represent a threat to the privacy of users. Unfortunately, all edge paradigms are open ecosystems, where multiple trust domains are controlled by different infrastructure owners. In such a context, it is not possible to know in advance if a certain service provider is trustworthy enough to respect the users' privacy. Therefore, this is a very serious threat that must be carefully considered.

There are various challenges in this area. First, personal data will be stored and processed by entities that are outside the control of the users. Therefore, it is essential to provide users with various efficient mechanisms that not only protect their information, but also allow users to query it and process it (e.g. auditable data, controlled disclosure). Second, it is necessary to achieve a balance between anonymity and responsibility. In this dynamic environment, users have the right to protect their identity and their personal data, but also have the responsibility to behave honestly. If a user misbehaves, it should be possible to use some mechanisms to identify the malicious party. Finally, we need to consider that human mobility is, in fact, quite predictable (cf. [70]): we usually go to the same places, follow the same routine every day. As a result, users will probably make use of the same edge data centers over and over. This poses a challenge to the development of privacy mechanisms that aim to protect the users' location and service usage.

*Virtualization.* The virtualization infrastructure is one of the core elements of edge paradigms, thus it is essential to protect it by designing and deploying security mechanisms in all edge data centers. Without these mechanisms, not only malicious insiders can take control of virtual machines deployed by users, but also malicious virtual machines can manipulate the services of edge data centers. There are numerous countermeasures that can be implemented in all commodity servers, such as isolation policies, hypervisor hardening, separation of roles and VMs, networking abstractions, and many others [68]. Note, however, that any mechanisms that depend on the restriction of physical access might be difficult to implement in this context.

*Fault tolerance and resilience.* No paradigm is ever going to be 100% secure and immune from threats, and edge paradigms are no exception. Misconfigurations, vulnerabilities, outdated software, and other weaknesses will allow malicious adversaries to disable or take control of certain elements of the whole infrastructure. It is then necessary to integrate various mechanisms and strategies (e.g. redundant operations, failover capabilities, disaster recovery mechanisms) that will allow the service infrastructure to continue its intended operation. However, the deployment of the edge data centers at the edge of the network is a double-edged sword. On the one hand, protection mechanisms can take advantage of the fact that various infrastructure providers might be available at the same location. On the other hand, as services are provided at a local level, there might be situations where no replacement is available.

*Forensics.* As we have already mentioned, no matter what protection mechanisms are put in place, edge paradigms will be

successfully attacked. These attacks will leave certain evidence behind, which can be used to reveal information about the attacker and his methods. The goal of forensics is to identify, recover, and preserve this evidence, so it can be presented in court. The management of evidence in edge paradigms is a very complex issue, mainly due to the existence of multiple actors, infrastructures, technologies, and scenarios. Nevertheless, it might be possible to make use of existing research in related areas, such as cloud forensics (cf. [71–73]), to solve certain issues such as mobile forensics, virtualization forensics, and storage forensics.

Besides, Wang et al. [17] and Zawoad et al. [74] have provided a detailed analysis of the main requirements of fog computing forensics and mobile cloud computing forensics, respectively. Both works agree that there are various common challenges in this area, such as (i) storing trusted evidence in a distributed ecosystem with multiple trust domains, (ii) respecting the privacy of other tenants when acquiring and managing evidence, and (iii) preserving the chain of custody of the evidence. Then again, both works agree that edge data centers should need less computational resources to manage potential evidence: due to their geographical location and their local scope, they do not manage as many resources (e.g. network traffic, virtual machines) as centralized cloud infrastructures.

## 5. Security challenges and opportunities

In the previous sections, we have reviewed the similarities and differences between all edge paradigms, and we have provided a detailed analysis on the threats that can target these paradigms—and the security mechanisms that should be used to protect them. In this section we will provide an analysis of the state of the art regarding security in all edge paradigms (Section 5.1), and we will conclude such analysis with a discussion on existing shortcomings and potential research areas (Section 5.2). As with Section 3.2, we will point out in our analysis potential synergies between all edge paradigms. Note, however, that we will also consider in our analysis other related paradigms (e.g. cloud computing, grid computing, peer-to-peer computing) and some of the enabling technologies that are used by edge paradigms (e.g. wireless networks, distributed and peer-to-peer systems, virtualization platforms [4]).

There are several reasons for this. Some of the underlying assumptions of the security mechanisms that were designed for related paradigms do not conflict with the requirements of edge paradigms. For example, certain peer-to-peer security mechanisms only require a decentralized infrastructure of peers that can communicate with each other. Other security protocols are independent of the underlying technologies that implement them, thus they can be easily adapted to other environments. Moreover, some security mechanisms were designed with a specific scenario in mind, but their functional elements can easily be mapped to edge paradigm scenarios. For example, the security components of certain trust management systems for grid computing only assume that servers hosted at different administrative domains can exchange trust information securely. These building blocks can be mapped to a edge paradigm scenario where multiple edge data centers that belong to different trust domains exchange information in a secure way. It is obvious that all these security mechanisms should not be adapted without an extensive analysis, yet they can prove that researchers do not need to start from scratch when designing security mechanisms for edge paradigms.

### 5.1. Specific challenges and promising solutions

#### 5.1.1. Identity and authentication

At present, there are no research works that analyze how to identify and authenticate the members of a world-wide infrastructure of interconnected edge data centers owned by different companies and individuals. Yet it might be possible to look for the solution to this problem in other related fields, such as federated cloud computing and peer-to-peer computing. In fact, there are multiple approaches that pursue the creation of inter-cloud identity management systems [75]. Such approaches make use of various standards, like SAML and OpenID, in order to provide Single-Sign On (SSO) authentication between clouds. As for peer-to-peer computing, there are also several mechanisms that provide mutual authentication without having to connect to a central authentication server [76]. As the design of these approaches is compatible with the underlying infrastructures of edge paradigms, all these approaches might be adapted to handle the authentication of edge data centers that belong to different trust domains.

On the other hand, there are some authentication infrastructures, which focus on user authentication within the same trust domain, explicitly designed for edge paradigms. For example, Donald et al. [77] defined a centralized infrastructure for MCC where a single trusted third party serves as the authentication server. However, this approach requires the authentication server to be accessible at all times, thus their applicability is limited. In another work, Ibrahim [78] developed a user authentication system that allows any fog user and fog node to mutually authenticate each other, yet this approach forces all fog nodes to store certain credential information of all the users of the trust domain. There are other works in the areas of MCC [79] and fog computing [12] that are able to authenticate users, even if the authentication server is not reachable, with less overhead. This is achieved by using pairing cryptosystems and secure hardware [79], or by using hybrid encryption (public-key and symmetric-key encryption) [12]. Although these mechanisms focus on user authentication, they might be useful for authenticating a federation of edge data centers that belong to the same trust domain.

Precisely, on the subject of user authentication, as edge data centers are located in the vicinity of end-users, researchers have proposed various authentication schemes that make use of location-specific information. For example, in the context of federated mobile cloud computing, Shouhuai et al. [80] introduced the concept of situational authentication, which is based on notions such as "whom you are with", "where you are", and "what time is it". Other authors, such as Bouzefrane [81], use Near Field Communication (NFC) to verify that a mobile device is offloading tasks to an authorized local cloudlet. Note that the notion of location-based authentication has been already studied in several other fields (e.g. wireless sensor networks [82], Internet of Things [83]), providing various mechanisms that could be adapted to edge paradigms.

As for user mobility, there have been some protocols that have tried to implement a secure and efficient handover authentication in MCC scenarios. For example, Yang et al. [84] provided an efficient design that allowed a mobile client to migrate from one region to another. Note that these protocols usually need to access an authentication server in a centralized cloud infrastructure, thus there is room for improvement. Finally, note that certain edge paradigms allow users to deploy their own personal data centers. Consequently, some works, such as the OPENi framework [85], have studied how to grant access to external users in such personal cloudlet platforms. In OPENi, the authentication component makes use of the OpenID Connect authentication layer, amongst other mechanisms. Therefore, the owner of the cloudlet decides which authentication servers he trusts, and what users are allowed to access the resources of the cloudlet.

#### 5.1.2. Access control systems

There are very few studies that have investigated the development of fine-grained access control mechanisms in the context of edge paradigms. One example is the OPENi framework [85], which

was mentioned in the previous section. In this framework, the authorization is based on OAuth 2.0, and the owner of the cloudlet defines the access rights of every resource by creating and storing access control lists (ACL) in a NoSQL database. This approach is more suitable for personal cloudlets, where their owners can define what operations can be performed on a resource by a certain user. Other approaches, like the one introduced by Huang et al. [86] and used by Stojmenovic et al. [12], use cryptographic primitives such as attribute-based encryption (ABE) to implement attribute-based access control policies. In this approach, users are provided with certain attributes, and access control rules connect such attributes with the operations that can be performed on a resource. This mechanism can be appropriate for a single trust domain, where service providers can use these attributes (alongside with their credentials) to get permission to deploy VMs in an edge data center.

Other authors have explored the deployment of policy enforcement components and the management of security policies. One simple example can be found in the architecture defined by Vassilakis et al. [87], which made use of a formal methodology to deploy security components in MEC small cells. These components provide protection and access control to various MEC services, such as radio resources and virtualization services. However, one of the most prominent examples is the policy management framework for fog computing designed by Dsouza et al. [88]. In this framework, the orchestration layer of the fog architecture is supported by a policy management module, which defines various components—including a repository of rules, an attribute database, and a session administrator. Moreover, the policies can be enforced at various levels, such as edge data centers, VM instances, and IoT devices. This policy management framework does not have any special architectural requirements beyond the existence of a core infrastructure, thus it can be applied to other paradigms such as mobile edge computing.

As for the existence of federated and distributed access control architectures in other related paradigms, there is actually an extensive literature on this subject [89,90]. Several of these mechanisms might be adapted to our context in order to solve existing open issues. For example, Almutairi et al. [91] developed a distributed access control architecture for multicloud environments, based on role-based access control (RBAC) policies, that also provided interdomain role mapping and constraint verification. This approach might be used to connect various entities that belong to different trust domains. Besides, there are other security mechanisms that, although not created for edge paradigms, might be suitable for certain scenarios. For example, the Direct Anonymous Attestation with Attributes (DAA-A) protocols allow anonymous users to prove that they possess a certain set of trusted attributes. These protocols can be implemented using the primitives defined in the Trusted Platform Module 2.0 (TPM 2.0) specification [92], thus they can be applied to scenarios where two edge data centers need to prove that they have certain attributes (e.g. location, capabilities) without disclosing their owners.

### 5.1.3. Protocol and network security

All the communication technologies that are used by edge paradigms are either mature standards (e.g. TCP/IP stack, Wi-Fi) or are being extensively studied by both industry and academia (e.g. 5G, Sigfox). They define their own security protocols and mechanisms, which are able to provide privacy and data integrity between two authenticated entities. One of the challenges in this area is the distribution of the credentials that will be used to negotiate the session keys. Yet solutions are available, even if more research is needed. For example, a designated certification authority controlled by one infrastructure provider can distribute credentials to all the elements located within his trust domain.

Cryptographic attributes, such as the attributes used in [86], can also be used as credentials in order to exchange session keys [93]. Besides, there are several works in various areas, such as federated content networks [94], that define how multiple trust domains can negotiate and maintain the interdomain credentials that will be used to establish secure channels. The requirements of these solutions are not very restrictive, thus they might be applied to edge paradigm environments.

Another aspect that we have to take into account is the security of the virtualized network infrastructure; that is, the network infrastructure that is used by the VMs deployed at edge data centers. As already pointed out by Ahmad et al. in their detailed analysis of the subject [95], both software defined networking (SDN) and network function virtualization (NFV) can be extremely useful in the context of edge paradigms. These approaches can be used in various ways, such as isolating different types of traffic even under adversarial conditions, isolating unsecure network devices, directing the traffic towards security devices, reconfiguring the systems in real time, etc. Notice that the original goal of NFV and SDN is to simplify the management of the network, by virtualizing the router functions and by implementing programmable network control and operation logic. These services are also beneficial for edge paradigms, as one of the challenges that need to be solved is the management of the network infrastructure [96,97]. Note, however, that both SDN and NFV have their own security challenges that need to be addressed [95,98].

### 5.1.4. Trust management

Although trust is one of the most important security requirements in edge paradigms, the amount of research that has been conducted in this area as of 2016 is quite limited. Actually, most of the research has focused only on the area of mobile cloud computing, analyzing the trust relationships between users. For example, Petri et al. [99] studied how various nodes could create a trustworthy peer-to-peer cloud, where feedback aggregation was used to identify egoist users. Also, Chen et al. [100] analyzed how call patterns can be used to derive the trust relationships between human users. One of the only works that explicitly analyzed how to calculate the reputation of edge data centers was developed by Hussain et al. [101]. In this work, the researchers describe the implementation of a centralized trust manager, which stores the reputation of LTE-deployed cloudlets. Using this system, users can rate the services of cloudlets anonymously.

Trust management has been a very active area of research in many other related fields [102–104]. Therefore, as with the other security properties, researchers might benefit from studying and adapting existing trust management systems. There are, in fact, several systems that might be applicable to edge paradigms, due to their focus on decentralized deployments and cross-domain relationships. One example is the self-managed trust management system by Kantert et al. [105]. In this work, autonomous servers from different administrative domains share their resources in a grid-like scenario. In contrast to other grid deployments, it is assumed that egoist or malicious servers will exist. Therefore, it is necessary to calculate a set of trust metrics in an autonomous and distributed way. This work might be used as a foundation for calculating trust values between edge data centers.

Another example is the quantitative trust management component, defined by Figueroa et al. and integrated into the Safety On Untrusted Network Devices (SOUND) platform [106]. This platform is comprised by several communities of trust, which contains various hosts. Whenever two hosts from different communities interact, they take into account not only their mutual trust, but also the trust between their communities, and the trust between the community and the other host. Due to the similarities between the communities of trust and the edge trusted domains, the design of

this particular component might be used as an input for the design of trust management systems deployed in edge data centers. Finally, Bennani et al. [107] defined a Bayesian network-based trust model for hybrid cloud computing environments. In this scenario, a private cloud can assess and track the reputation of various services provided by public clouds. This kind of approach might be used to track the reputation of services that are available to the whole ecosystem, such as Security-as-a-Service solutions.

### 5.1.5. Intrusion detection systems

Most of the research on the area of intrusion detection and prevention systems has focused on mobile cloud computing, with only a few exceptions such as the active honeypot system designed by Mtibaa et al. [108] that focused on detecting local adversaries in mobile edge computing deployments. Yet some of these MCC-centric research works might be used for other paradigms, too. Gai et al. [109] proposed a framework where mobile devices using 5G networks could delegate their intrusion detection tasks to centralized services located in the cloud. While this research was focused on centralized cloud services, it might be possible to adapt this framework to a more distributed approach, where the IDS services will be deployed in nearly located edge data centers. Such services will then have a comprehensive view of the state of their surroundings. Also, Shi et al. [110] presented a distributed IDS deployed in a cloudlet mesh architecture. In this architecture, the members of the cloudlet can collaborate with each other and with external entities in order to detect malware, malicious attacks, and others. This type of collaborative IDS might also be used by a federation of edge data centers to monitor the traffic of a certain geographical location.

Although there is still work to be done, it is perfectly possible to reuse various IDS mechanisms and solutions developed for cloud computing [111] and other related paradigms. The reason is simple. The main task of edge data centers is to provide cloud computing capabilities to users. Therefore, edge data centers can benefit from IDS that monitor the behavior of VMs, the internal network and their surroundings. The main challenge here is to deal with the distributed nature of the whole infrastructure, where multiple trust domains coexist. Yet many IDS solutions do not need of centralized infrastructures—they can monitor their environment autonomously. One example is the CROW solution, developed by Pitropakis et al. [112]. This IDS solution makes use of the computational power of GPU cards to effectively monitor the health of each VMs, detecting both attacks against the infrastructure and the presence of malicious insiders. Other examples are the IDS solutions that rely on software based networking (SDN) principles, and provide services such as deep packet inspection, network reconfiguration, policy management, and flow-based anomaly detection, amongst others (cf. [95]).

Moreover, there are actually various IDS frameworks whose goal is to interconnect and monitor different trust domains. Elements of these frameworks might be reused or adapted to our context. For example, Luo et al. [113] introduced a security architecture for federated cloud environments that facilitates the early detection of cyberattacks and the deployment of early warning systems such as honeypots. Instances of this architecture need to be deployed in the centralized command and control center of every trust domain, thus its applicability to a N-tiered hierarchy needs to be further studied. Yet the architecture also introduces various mechanisms that allow multiple trust domains to coordinate in-cloud and cross-cloud defense activities. Finally, some studies, like [114], have provided an analysis of attacks that specifically target federated cloud environments—and that can also target edge paradigms.

### 5.1.6. Privacy

In the field of edge paradigms, privacy is one area that has been particularly active in the last years. In fact, many of the security protocols presented in the previous sections (e.g. entity authentication [79] and authorization [12,92], trust management [101]) allow users to interact with edge data centers and other entities in an anonymous way. Besides, there is a multitude of data privacy mechanisms specifically developed for the mobile cloud computing paradigm. These mechanisms tackle several challenges such as enforcing privacy policies when migrating code and data amongst collaborating mobile devices [115], and concealing the location of a set of clients that are located in the same geographical area by means of establishing a peer-to-peer network [116]. These mechanisms are designed for a collaborative cloud of local devices, yet they only require that all devices are interconnected and know their physical location. Therefore, they might also provide some inputs on the design of future privacy mechanisms for collaborative edge data centers. Note that there are other mechanisms, such as the software-defined pseudonym system for vehicular networks developed by Huang et al. [117], that make full use of the concept of interconnected local cloudlets.

Moreover, privacy has been one of the most researched fields in cloud computing [118]. There are various cloud computing processes that have been enhanced with privacy features, such as protecting VMs during their storage and execution, and migrating VMs from one data center to another [119]. Most of these solutions do not need a centralized infrastructure, and only require of a Trusted Platform Module (TPM), thus they can be implemented in the commodity servers that are available in edge data centers. Moreover, there are specific privacy mechanisms, such as data encryption, secure data sharing, encrypted data search, integrity verification, and many others [120], whose main goal is to protect the personal data of users. Some of these mechanisms do not have high computational requirements, thus they can be implemented in the user devices that interact with the edge data centers.

Notice that there are some use cases (e.g. personal cloudlets, corporate environments) where there is a trust relationship between the users and the edge data centers located at their vicinity. In such cases, it is possible to deploy privacy helper entities in the edge data centers. These entities will act as a front-end for the users, and can implement various data privacy mechanisms. These mechanisms can be used to control the quality and granularity of the personal information that is received by service providers or other remote entities (cf. [121,122]). In addition, the privacy helpers can implement other privacy services, such as protecting the users' identities from other remote services by creating pseudonyms and/or concealing their addresses (cf. [11]). Finally, it should be noted that the edge paradigms themselves can actually be used to strengthen the privacy features of certain services, such as crowdsourcing. For example, Abdo et al. [123] demonstrated that, by deploying a crowdsourcing platform in a trusted edge data center, it was possible to protect the anonymity of the participants of certain location-based services.

### 5.1.7. Virtualization

In the context of cloud computing, the security of virtualization infrastructures is a field that has been intensively studied in recent years [124]. Fortunately, many secure virtualization mechanisms do not need centralized managers or specific hardware unavailable to commodity servers. Therefore, they can be applied to the virtualization infrastructures that are used in edge paradigms. One clear example is the notion of Virtual Trusted Platform Modules (vTPM) [125]. By virtualizing Trusted Platform Modules (TPM), vTPMs are able to provide TPM services (e.g. secure storage, cryptographic functions) to any virtual machine that is running on top of a hypervisor. In fact, existing hypervisor platforms, such

**Table 5**
State of the art in edge security as of Q3 2016.

| | Fog computing | MEC | MCC | Other paradigms |
|---|---|---|---|---|
| Identity and authentication | [12,78] | – | [77,79–81,84,85] | [75,76,82,83] |
| Access control systems | [86,12,88] | [87] | [85] | [89–92] |
| Protocol and network security | – | – | – | [86,93–97] |
| Trust management | – | – | [99–101] | [102–107] |
| Intrusion detection systems | – | [108] | [109,110] | [112,95,113,114] |
| Privacy | [12] | – | [79,101,115,116,121,11,123,117] | [92,118–120,122] |
| Virtualization | – | – | [129–131] | [124] |
| Forensics | [17] | – | [74] | [132] |

as Xen and Hyper-V, already provide support for vTPMs. These services have been used to implement various security services that are relevant to edge paradigms, such as VM creation and cloning [126], VM migration [127], platform attestation [128], and many others (data storage, secure rollbacks).

Besides, in the area of mobile cloud computing, there are some research studies that propose secure computation offloading solutions. For example, Hao et al. [129] proposed a system that allows a subset of a mobile application to securely run in a cloud server. Also, Dhanya et al. [130] proposed a secure partitioning mechanism that kept the most sensitive or vulnerable parts of an application in the mobile device. These solutions might be adapted to other edge paradigms, as there might be some cases where a VM only needs to send a small agent to other data centers (cf. [131]).

### 5.2. Summary

Table 5 provides a summary of the state of the art that was reviewed in the previous section. In this table, all studies are classified according to the original paradigm for which they were designed. One obvious conclusion is that there are very few studies that have been specifically designed for fog computing and mobile edge computing, compared to the amount of studies that have been focused on mobile cloud computing. The reasons are simple: (i) these paradigms were created very recently, and their infrastructure has not been fully defined, and (ii) the mobile cloud computing paradigm has been studied longer. The reader should note, however, that many studies in the area of mobile cloud computing have not targeted the security of edge data centers (i.e. cloudlets), but distributed clusters of mobile devices instead.

Even if the number of studies that target edge paradigms is quite limited, it does not mean that researchers must start from zero when developing new security mechanisms. As we have seen in the last section, it might be possible to use the security mechanisms and components that have been designed for other related paradigms as a foundation for the development of novel edge security mechanisms. Moreover, we also have shown in the most recent section that it might possible to reuse or adapt various security mechanisms that were specifically designed for one edge paradigm to the other edge paradigms. However, it is necessary to analyze how the specific nuances of every edge paradigm – like underlying features of mobile network operator infrastructures or user-owned edge data centers – will affect this adaptation process.

Having said this, several issues will need to be studied and evaluated in the near future. Some examples of these issues are explained here briefly. It is necessary to investigate the impact that certain attacks, such as denial of service, rogue data centers and malicious VMs, will have on the service infrastructure. In addition, it must be assessed how such attacks can be detected and neutralized by intrusion detection/prevention systems. The edge paradigm ecosystem must provide support for various identity management frameworks, including those used by prominent application scenarios like the Internet of Things. It must be possible for administrators to maintain a consistent network configuration and access control policy across all elements of the edge infrastructure with as little overhead as possible. Besides, we need to analyze how trust management systems can benefit other security mechanisms as well as the exact impact that edge paradigms will have on the privacy of their users. It is essential to reduce the latency of all security mechanisms as much as possible, and to study the security of mobile entities in this context.

Furthermore, there are certain research areas that have been neglected in the context of edge paradigms, such as secure software engineering, security and usability, fault tolerance and resilience, and forensics. All of them are essential in this context. By considering the specific features of edge paradigms (e.g. context awareness or interaction with mobile clients) during the development of security-aware software systems, the vulnerabilities specific to our context will be greatly reduced. Usability is another essential factor, as the development of usable security mechanisms will limit misconfigurations and facilitate the maintenance of the whole ecosystem. Thanks to fault tolerance, the service infrastructure will be able to continue its operation, even if at a reduced level. Last, malicious adversaries can be identified and prosecuted if effective forensics procedures are in place.

## 6. Conclusions

In this study, we have analyzed from a holistic perspective the security threats and challenges that affect edge paradigms, such as fog computing, mobile edge computing, and mobile cloud computing. In the first part of our analysis, we identified the features and problems that are common to all edge paradigms. In the second part, we provided a novel analysis of the multiple threats that target all edge paradigms, alongside a detailed study regarding the state of the art of security mechanisms that should be integrated into all edge paradigms. As a conclusion of this analysis, we have shown that research should not be compartmentalized, but all edge paradigms should consider the advances in other paradigms. Nevertheless, the security of edge paradigms is still in its infancy; thus, there are multiple open issues that merit consideration in the near future.

## References

[1] National Institute of Standards and Technology, The NIST Definition of Cloud Computing, SP 800-145, 2011. http://csrc.nist.gov/publications/PubsSPs.html#800-145 [Online; accessed 15.09.16].

[2] International Data Corporation, IDC, Worldwide Public Cloud Services Spending Forecast to Double by 2019, According to IDC, 2016. https://www.idc.com/getdoc.jsp?containerId=prUS40960516 [Online; accessed 15.09.16].

[3] M. Satyanarayanan, A brief history of cloud offload: A personal journey from Odyssey through cyber foraging to cloudlets, Mob. Comput. Commun. 18 (4) (2015) 19–23. http://dx.doi.org/10.1145/2721914.2721921.

[4] L.M. Vaquero, L. Rodero-Merino, Finding your way in the fog: Towards a comprehensive definition of fog computing, SIGCOMM Comput. Commun. Rev. 44 (5) (2014) 27–32. http://dx.doi.org/10.1145/2677046.2677052.

[5] M.T. Beck, M. Maier, Mobile edge computing: Challenges for future virtual network embedding algorithms, in: Proceedings of the 8th International Conference on Advanced Engineering Computing and Applications in Sciences, ADVCOMP, 2014, pp. 65–70.

[6] Y. Wang, I.-R. Chen, D.-C. Wang, A survey of mobile cloud computing applications: Perspectives and challenges, Wirel. Pers. Commun. 80 (4) (2015) 1607–1623. http://dx.doi.org/10.1007/s11277-014-2102-7.

[7] F. Manco, J. Martins, K. Yasukata, J. Mendes, S. Kuenzer, F. Huici, The case for the superfluid cloud, in: Proceedings of the 7th USENIX Conference on Hot Topics in Cloud Computing, HotCloud, 2015, pp. 1–6. URL http://dl.acm.org/citation.cfm?id=2827719.2827726.

[8] P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, E. Riviere, Edge-centric Computing: Vision and challenges, SIGCOMM Comput. Commun. Rev. 45 (5) (2015) 37–42. http://dx.doi.org/10.1145/2831347.2831354.

[9] OPENi Consortium, Deliverable 2.3 - Security and Privacy Considerations for Cloud-based Services and Cloudlets, 2013. http://www.openi-ict.eu/deliverables/ [Online; accessed 15.09.16].

[10] H. Suo, Z. Liu, J. Wan, K. Zhou, Security and privacy in mobile cloud computing, in: Proceedings of the 9th International Wireless Communications and Mobile Computing Conference, IWCMC, 2013, pp. 655–659. http://dx.doi.org/10.1109/IWCMC.2013.6583635.

[11] H. Takabi, S.T. Zargar, J.B.D. Joshi, Mobile cloud computing and its security and privacy challenges, in: Cloud Technology: Concepts, Methodologies, Tools, and Applications, IGI Global, 2015, pp. 1561–1584. http://dx.doi.org/10.4018/978-1-4666-6539-2.ch073.

[12] I. Stojmenovic, S. Wen, X. Huang, H. Luan, An overview of fog computing and its security issues, Concurr. Comput.: Pract. Exper. 28 (10) (2016) 2991–3005.

[13] K. Lee, D. Kim, D. Ha, U. Rajput, H. Oh, On security and privacy issues of fog computing supported Internet of things environment, in: Proceedings of the 6th International Conference on the Network of the Future, NOF, 2015, pp. 1–3. http://dx.doi.org/10.1109/NOF.2015.7333287.

[14] S. Yi, Z. Qin, Q. Li, Security and privacy issues of fog computing: A survey, in: K. Xu, H. Zhu (Eds.), Wireless Algorithms, Systems, and Applications, in: Lecture Notes in Computer Science, vol. 9204, Springer International Publishing, 2015, pp. 685–695. http://dx.doi.org/10.1007/978-3-319-21837-3_67.

[15] Yucianga Ltd, Open Fog Computing and Mobile Edge Cloud Gain Momentum, 2015. http://yucianga.info/?p=938 [Online; accessed 15.09.16].

[16] I. Stojmenovic, S. Wen, The fog computing paradigm: Scenarios and security issues, in: Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, FedCSIS, 2014, pp. 1–8. http://dx.doi.org/10.15439/2014F503.

[17] Y. Wang, T. Uehara, R. Sasaki, Fog computing: Issues and challenges in security and forensics, in: Proceedings of the 39th IEEE Annual Computer Software and Applications Conference, COMPSAC, Vol. 3, 2015, pp. 53–59. http://dx.doi.org/10.1109/COMPSAC.2015.173.

[18] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog computing and its role in the Internet of things, in: Proceedings of the 1st Edition of the MCC Workshop on Mobile Cloud Computing, 2012, pp. 13–16. http://dx.doi.org/10.1145/2342509.2342513.

[19] F. Bonomi, R. Milito, P. Natarajan, J. Zhu, Fog computing: A platform for Internet of things and analytics, in: N. Bessis, C. Dobre (Eds.), Big Data and Internet of Things: A Roadmap for Smart Environments, in: Studies in Computational Intelligence, vol. 546, Springer International Publishing, 2014, pp. 169–186. http://dx.doi.org/10.1007/978-3-319-05029-4_7.

[20] S. Yi, C. Li, Q. Li, A survey of fog computing: Concepts, applications and issues, in: Proceedings of the 2015 Workshop on Mobile Big Data, Mobidata, 2015, pp. 37–42. http://dx.doi.org/10.1145/2757384.2757397.

[21] T.H. Luan, L. Gao, Z. Li, Y. Xiang, L. Sun, Fog computing: Focusing on mobile users at the edge. Preprint, available online arXiv:1502.01815.

[22] J. Zao, T.T. Gan, C.K. You, S. Rodriguez Mendez, C.E. Chung, Y.T. Wang, T. Mullen, T.P. Jung, Augmented brain computer interaction based on fog computing and linked data, in: Proceedings of the International Conference on Intelligent Environments, IE, 2014, pp. 374–377. http://dx.doi.org/10.1109/IE.2014.54.

[23] K. Ha, Z. Chen, W. Hu, W. Richter, P. Pillai, M. Satyanarayanan, Towards wearable cognitive assistance, in: Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys, 2014, pp. 68–81. http://dx.doi.org/10.1145/2594368.2594383.

[24] I. Stojmenovic, Fog computing: A cloud to the ground support for smart things and machine-to-machine networks, in: Proceedings of the 2014 Australasian Telecommunication Networks and Applications Conference, ATNAC, 2014, pp. 117–122. http://dx.doi.org/10.1109/ATNAC.2014.7020884.

[25] S. Jingtao, L. Fuhong, Z. Xianwei, L. Xing, Steiner tree based optimal resource caching scheme in fog computing, China Commun. 12 (8) (2015) 161–168. http://dx.doi.org/10.1109/CC.2015.7224698.

[26] O.T.T. Kim, N.D. Tri, V.D. Nguyen, N. Tran, C.S. Hong, A shared parking model in vehicular network using fog and cloud environment, in: Proceedings of the 17th Asia-Pacific Network Operations and Management Symposium, APNOMS, 2015, pp. 321–326. http://dx.doi.org/10.1109/APNOMS.2015.7275447.

[27] Open Fog Consortium, http://www.openfogconsortium.org/ [Online; accessed 15.09.16].

[28] W.S. Chin, H. soo Kim, Y.J. Heo, J.W. Jang, A context-based future network infrastructure for IoT services, Procedia Comput. Sci. 56 (2015) 266–270. http://dx.doi.org/10.1016/j.procs.2015.07.207.

[29] S. Datta, C. Bonnet, J. Haerri, Fog computing architecture to enable consumer centric Internet of things services, in: Proceedings of the 2015 IEEE International Symposium on Consumer Electronics, ISCE, 2015, pp. 1–2. http://dx.doi.org/10.1109/ISCE.2015.7177778.

[30] M. Zhanikeev, A cloud visitation platform to facilitate cloud federation and fog computing, Computer 48 (5) (2015) 80–83. http://dx.doi.org/10.1109/MC.2015.122.

[31] S.W. Loke, The Internet of Flying-Things: Opportunities and Challenges with Airborne Fog Computing and Mobile Cloud in the Clouds. Preprint, available online arXiv:1507.04492. URL http://arxiv.org/abs/1507.04492.

[32] IBM News Releases, IBM and Nokia Siemens Networks announce world first mobile edge computing platform, 2013. http://www-03.ibm.com/press/us/en/pressrelease/40490.wss [Online; accessed 15.09.16].

[33] ETSI, Mobile-Edge Computing Introductory Technical White Paper, 2014. http://www.etsi.org/technologies-clusters/technologies/mobile-edge-computing [Online; accessed 15.09.16].

[34] J.O. Fajardo Portillo, I. Taboada Puente, F. Liberal Malaina, Radio-aware service-level scheduling to minimize downlink traffic delay through mobile edge computing, in: Proceedings of the 7th EAI International Conference on Mobile Networks and Management, MONAMI, 2015, pp. 1–14.

[35] Y.C. Hu, M. Patel, D. Sabella, N. Sprecher, V. Young, Mobile Edge Computing: A key technology towards 5G, 2015. http://www.etsi.org/technologies-clusters/technologies/mobile-edge-computing [Online; accessed 15.09.16].

[36] A.J. Staring, G. Karagiannis, Cloud computing models and their application in LTE based cellular systems, in: Proceedings of the 2013 IEEE International Conference on Communications Workshops, ICC, 2013, pp. 750–755. http://dx.doi.org/10.1109/ICCW.2013.6649333.

[37] M.A. Puente, Z. Becvar, M. Rohlik, F. Lobillo, E. Calvanese Strinati, A seamless integration of computationally-enhanced base stations into mobile networks towards 5G, in: Proceedings of the IEEE 81st Vehicular Technology Conference, VTC Spring, 2015, pp. 1–5. http://dx.doi.org/10.1109/VTCSpring.2015.7145645.

[38] M. Maier, B.P. Rimal, Invited paper: The audacity of fiber-wireless (FiWi) networks: Revisited for clouds and cloudlets, China Commun. 12 (8) (2015) 33–45. http://dx.doi.org/10.1109/CC.2015.7224704.

[39] M. Ali, Green cloud on the horizon, in: M. Jaatun, G. Zhao, C. Rong (Eds.), Cloud Computing, in: Lecture Notes in Computer Science, vol. 5931, Springer, Berlin, Heidelberg, 2009, pp. 451–459. http://dx.doi.org/10.1007/978-3-642-10665-1_41.

[40] P. Bahl, R.Y. Han, L.E. Li, M. Satyanarayanan, Advancing the state of mobile cloud computing, in: Proceedings of the 3rd ACM Workshop on Mobile Cloud Computing and Services, MCS, 2012, pp. 21–28. http://dx.doi.org/10.1145/2307849.2307856.

[41] M.R. Rahimi, J. Ren, C.H. Liu, A.V. Vasilakos, N. Venkatasubramanian, Mobile cloud computing: A survey, state of art and future directions, Mob. Netw. Appl. 19 (2) (2014) 133–143. http://dx.doi.org/10.1007/s11036-013-0477-4.

[42] H.T. Dinh, C. Lee, D. Niyato, P. Wang, A survey of mobile cloud computing: Architecture, applications, and approaches, Wirel. Commun. Mob. Comput. 13 (18) (2013) 1587–1611. http://dx.doi.org/10.1002/wcm.1203.

[43] C.S. Magurawalage, K. Yang, K. Wang, Aqua Computing: Coupling Computing and Communications. Preprint, available online arXiv:1510.07250. URL http://arxiv.org/abs/1510.07250.

[44] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, R. Buyya, Cloud-based augmentation for mobile devices: Motivation, taxonomies, and open challenges, IEEE Commun. Surv. Tutor. 16 (1) (2014) 337–368. http://dx.doi.org/10.1109/SURV.2013.070813.00285.

[45] M. Satyanarayanan, P. Bahl, R. Caceres, N. Davies, The case for VM-based cloudlets in mobile computing, IEEE Pervasive Comput. 8 (4) (2009) 14–23. http://dx.doi.org/10.1109/MPRV.2009.82.

[46] M. Satyanarayanan, et al. Elijah: Cloudlet-based Mobile Computing, 2015. http://elijah.cs.cmu.edu/ [Online; accessed 15.09.16].

[47] Y. Gao, W. Hu, K. Ha, B. Amos, P. Pillai, M. Satyanarayanan, Are Cloudlets Necessary?, Technical Report CMU–CS–15–139, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 2015, October.

[48] E.E. Marinelli, Hyrax: Cloud computing on mobile devices using mapreduce (Master Thesis CMU–CS–09–164), School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 2009, September.

[49] K. Habak, M. Ammar, K. Harras, E. Zegura, Femto clouds: Leveraging mobile devices to provide cloud service at the edge, in: Proceedings of the IEEE 8th International Conference on Cloud Computing, CLOUD, 2015, pp. 9–16. http://dx.doi.org/10.1109/CLOUD.2015.12.

[50] R. Hasan, M. Hossain, R. Khan, Aura: An IoT based cloud infrastructure for localized mobile computation outsourcing, in: Proceedings of the 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, MobileCloud, 2015, pp. 183–188. http://dx.doi.org/10.1109/MobileCloud.2015.37.

[51] E. Kavvadia, S. Sagiadinos, K. Oikonomou, G. Tsioutsiouliklis, S. Assa, Elastic virtual machine placement in cloud computing network environments, Comput. Netw. 93 (3) (2015) 435–447. http://dx.doi.org/10.1016/j.comnet.2015.09.038.

[52] J. Oueis, E. Strinati, S. Barbarossa, The fog balancing: Load distribution for small cell cloud computing, in: Proceedings of the IEEE 81st Vehicular Technology Conference, VTC Spring, 2015, pp. 1–6. http://dx.doi.org/10.1109/VTCSpring.2015.7146129.

[53] B. Ottenwälder, B. Koldehofe, K. Rothermel, U. Ramachandran, MigCEP: Operator migration for mobility driven distributed complex event processing, in: Proceedings of the 7th ACM International Conference on Distributed Event-based Systems, DEBS, 2013, pp. 183–194. http://dx.doi.org/10.1145/2488222.2488265.

[54] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, B. Koldehofe, Opportunistic spatio-temporal event processing for mobile situation awareness, in: Proceedings of the 7th ACM International Conference on Distributed Event-based Systems, DEBS, 2013, pp. 195–206. http://dx.doi.org/10.1145/2488222.2488266.

[55] B. Ottenwälder, B. Koldehofe, K. Rothermel, K. Hong, D. Lillethun, U. Ramachandran, MCEP: A mobility-aware complex event processing system, ACM Trans. Internet Technol. 14 (1) (2014) 6:1–6:24. http://dx.doi.org/10.1145/2633688.

[56] P. Paglierani, High performance computing and network function virtualization: A major challenge towards network programmability, in: Proceedings of the 2015 IEEE International Black Sea Conference on Communications and Networking, BlackSeaCom, 2015, pp. 137–141. http://dx.doi.org/10.1109/BlackSeaCom.2015.7185102.

[57] K. Romer, B. Ostermaier, F. Mattern, M. Fahrmair, W. Kellerer, Real-time search for real-world entities: A survey, Proc. IEEE 98 (11) (2010) 1887–1902. http://dx.doi.org/10.1109/JPROC.2010.2062470.

[58] S. Abedin, M. Alam, N. Tran, C.S. Hong, A fog based system model for cooperative IoT node pairing using matching theory, in: Proceedings of the 17th Asia-Pacific Network Operations and Management Symposium, APNOMS, 2015, pp. 309–314. http://dx.doi.org/10.1109/APNOMS.2015.7275445.

[59] P. Jayaraman, J. Gomes, H. Nguyen, Z. Abdallah, S. Krishnaswamy, A. Zaslavsky, CARDAP: A scalable energy-efficient context aware distributed mobile data analytics platform for the fog, in: Y. Manolopoulos, G. Trajcevski, M. Kon-Popovska (Eds.), Advances in Databases and Information Systems, in: Lecture Notes in Computer Science, vol. 8716, Springer International Publishing, 2014, pp. 192–206. http://dx.doi.org/10.1007/978-3-319-10933-6_15.

[60] M. Aazam, E.-N. Huh, E-HAMC: Leveraging fog computing for emergency alert service, in: Proceedings of the 2015 IEEE International Conference on Pervasive Computing and Communication Workshops, 2015, pp. 518–523. http://dx.doi.org/10.1109/PERCOMW.2015.7134091.

[61] N. Nikaein, E. Schiller, R. Favraud, K. Katsalis, D. Stavropoulos, I. Alyafawi, Z. Zhao, T. Braun, T. Korakis, Network store: Exploring slicing in future 5G networks, in: Proceedings of the 10th International Workshop on Mobility in the Evolving Internet Architecture, MobiArch, 2015, pp. 8–13. http://dx.doi.org/10.1145/2795381.2795390.

[62] D. Miessler, Securing the Internet of things: Mapping IoT attack surface areas with the OWASP IoT top 10 project, in: Proceedings of the 2015 RSA Conference, 2015.

[63] R. Roman, J. Zhou, J. Lopez, On the features and challenges of security and privacy in distributed Internet of things, Comput. Netw. 57 (10) (2013) 2266–2279. http://dx.doi.org/10.1016/j.comnet.2012.12.018.

[64] K.-K.R. Choo, Cloud computing: Challenges and future directions, Trends Issues Crime Criminal Justice 400 (2010) 1–6.

[65] S. Landau, Highlights from making sense of Snowden, Part II: What's significant in the NSA revelations, IEEE Secur. Privacy 12 (1) (2014) 62–64. http://dx.doi.org/10.1109/MSP.2013.161.

[66] N.V. Juliadotter, K.-K.R. Choo, Cloud attack and risk assessment taxonomy, IEEE Cloud Comput. 2 (1) (2015) 14–20. http://dx.doi.org/10.1109/MCC.2015.2.

[67] T. Lengyel, T. Kittel, J. Pfoh, C. Eckert, Multi-tiered security architecture for ARM via the virtualization and security extensions, in: Proceedings of the 25th International Workshop on the Database and Expert Systems Applications DEXA, 2014, pp. 308–312. http://dx.doi.org/10.1109/DEXA.2014.68.

[68] G. Pék, L. Buttyán, B. Bencsáth, A survey of security issues in hardware virtualization, ACM Comput. Surv. 45 (3) (2013) 40:1–40:34. http://dx.doi.org/10.1145/2480741.2480757.

[69] V. Vassilakis, E. Panaousis, H. Mouratidis, Security challenges of small cell as a service in virtualized mobile edge computing environments, in: 10th IFIP International Conference on Information Security Theory and Practice, WISTP 2016, Springer International Publishing, 2016, pp. 70–84. http://dx.doi.org/10.1007/978-3-319-45931-8_5.

[70] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (5968) (2010) 1018–1021. http://dx.doi.org/10.1126/science.1177170.

[71] Q. Do, B. Martini, K.-K.R. Choo, A cloud-focused mobile forensics methodology, IEEE Cloud Comput. 2 (4) (2015) 60–65. http://dx.doi.org/10.1109/MCC.2015.71.

[72] D. Barrett, G. Kipper, Virtualization and Forensics: A Digital Forensic Investigator's Guide to Virtual Environments, Elsevier, 2010.

[73] N.H. Ab Rahman, N.D.W. Cahyani, K.-K.R. Choo, Cloud incident handling and forensic-by-design: Cloud storage as a case study, Concurr. Comput. Pract. Exper. (2016) http://dx.doi.org/10.1002/cpe.3868, in press.

[74] S. Zawoad, R. Hasan, Towards a systematic analysis of challenges and issues in secure mobile cloud forensics, in: Proceedings of the 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (Mobile-Cloud), 2015, pp. 237–238. http://dx.doi.org/10.1109/MobileCloud.2015.32.

[75] A.N. Toosi, R.N. Calheiros, R. Buyya, Interconnected cloud computing environments: Challenges, taxonomy, and survey, ACM Comput. Surv. 47 (1) (2014) 7:1–7:47. http://dx.doi.org/10.1145/2593512.

[76] D.S. Touceda, J.M.S. Cmara, S. Zeadally, M. Soriano, Attribute-based authorization for structured Peer-to-Peer (P2P) networks, Comput. Stand. Interfaces 42 (2015) 71–83. http://dx.doi.org/10.1016/j.csi.2015.04.007.

[77] A. Donald, L. Arockiam, A secure authentication scheme for mobicloud, in: International Conference on Computer Communication and Informatics, ICCCI, 2015, pp. 1–6. http://dx.doi.org/10.1109/ICCCI.2015.7218101.

[78] M.H. Ibrahim, Octopus: An edge-fog mutual authentication scheme, Int. J. Netw. Secur. 18 (6) (2016) 1089–1101.

[79] J.-L. Tsai, N.-W. Lo, A privacy-aware authentication scheme for distributed mobile cloud computing services, IEEE Syst. J. 9 (3) (2015) 805–815. http://dx.doi.org/10.1109/JSYST.2014.2322973.

[80] S. Xu, E.P. Ratazzi, W. Du, Security architecture for federated mobile cloud computing, in: Mobile Cloud Security, Springer, 2016, (in press).

[81] S. Bouzefrane, A. Benkara Mostefa, F. Houacine, H. Cagnon, Cloudlets authentication in nfc-based mobile computing, in: Proceedings of the 2nd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2014, pp. 267–272. http://dx.doi.org/10.1109/MobileCloud.2014.46.

[82] Y. Zeng, J. Cao, J. Hong, S. Zhang, L. Xie, Secure localization and location verification in wireless sensor networks: A survey, J. Supercomput. 64 (3) (2010) 685–701. http://dx.doi.org/10.1007/s11227-010-0501-4.

[83] K. Habib, W. Leister, Context-aware authentication for the Internet of things, in: Proceedings of the 11th International Conference on Autonomic and Autonomous Systems, ICAS, 2015, pp. 134–139.

[84] X. Yang, X. Huang, J.K. Liu, Efficient handover authentication with user anonymity and untraceability for mobile cloud computing, Future Gener. Comput. Syst. (2015) http://dx.doi.org/10.1016/j.future.2015.09.028, URL http://www.sciencedirect.com/science/article/pii/S0167739X15003088.

[85] D. McCarthy, P. Malone, J. Hange, K. Doyle, E. Robson, D. Conway, S. Ivanov, L. Radziwonowicz, R. Kleinfeld, T. Michalareas, T. Kastrinogiannis, N. Stasinos, F. Lampathaki, Personal cloudlets: Implementing a user-centric datastore with privacy aware access control for cloud-based data platforms, in: Proceedings of the IEEE/ACM 1st International Workshop on Technical and Legal aspects of data privacy and Security, TELERISE, 2015, pp. 38–43. http://dx.doi.org/10.1109/TELERISE.2015.15.

[86] X. Huang, Y. Xiang, E. Bertino, J. Zhou, L. Xu, Robust multi-factor authentication for fragile communications, IEEE Trans. Dependable Secure Comput. 11 (6) (2014) 568–581. http://dx.doi.org/10.1109/TDSC.2013.2297110.

[87] V. Vassilakis, I.P. Chochliouros, A.S. Spiliopoulou, E. Sfakianakis, M. Belesioti, N. Bompetsis, M. Wilson, C. Turyagyenda, A. Dardamanis, Security analysis of mobile edge computing in virtualized small cell networks, in: 12th IFIP International Conference on Artificial Intelligence Applications and Innovations, AIAI 2016, Springer International Publishing, 2016, pp. 653–665. http://dx.doi.org/10.1007/978-3-319-44944-9_58.

[88] C. Dsouza, G.-J. Ahn, M. Taguinod, Policy-driven security management for fog computing: Preliminary framework and a case study, in: Proceedings of the IEEE 15th International Conference on Information Reuse and Integration, IRI, 2014, pp. 16–23. http://dx.doi.org/10.1109/IRI.2014.7051866.

[89] H. Li, S. Wang, X. Tian, W. Wei, C. Sun, A Survey of extended role-based access control in cloud computing, in: W. E. Wong (Ed.), Proceedings of the 4th International Conference on Computer Engineering and Networks, 2015, pp. 821–831. http://dx.doi.org/10.1007/978-3-319-11104-9_95.

[90] W. Elsayed, T. Gaber, N. Zhang, M. Ibrahim Moussa, Access control models for pervasive environments: A survey, in: T. Gaber, A.E. Hassanien, N. El-Bendary, N. Dey (Eds.), Proceedings of the 1st International Conference on Advanced Intelligent System and Informatics, AISI, in: Advances in Intelligent Systems and Computing, vol. 407, Springer International Publishing, 2015, pp. 511–522. http://dx.doi.org/10.1007/978-3-319-26690-9_45.

[91] A.A. Almutairi, M.I. Sarfraz, S. Basalamah, W.G. Aref, A. Ghafoor, A distributed access control architecture for cloud computing, IEEE Softw. 29 (2) (2012) 36–44. http://dx.doi.org/10.1109/MS.2011.153.

[92] L. Chen, R. Urian, DAA-A: Direct anonymous attestation with attributes, in: M. Conti, M. Schunter, I. Askoxylakis (Eds.), Trust and Trustworthy Computing, in: Lecture Notes in Computer Science, vol. 9229, Springer International Publishing, 2015, pp. 228–245. http://dx.doi.org/10.1007/978-3-319-22846-4_14.

[93] M.C. Gorantla, C. Boyd, J.M. González Nieto, Attribute-based authenticated key exchange, in: Proceedings of the 15th Australasian Conference on Information Security and Privacy, ACISP, Springer, Berlin Heidelberg, 2010, pp. 300–317. http://dx.doi.org/10.1007/978-3-642-14081-5_19.

[94] H.M. Pimentel, S. Kopp, M.A.S. Jr., R.M. Silveira, G. Bressan, OCP: A protocol for secure communication in federated content networks, Comput. Commun. 68 (2015) 47–60. http://dx.doi.org/10.1016/j.comcom.2015.07.026, security and Privacy in Unified Communications: Challenges and Solutions.

[95] I. Ahmad, S. Namal, M. Ylianttila, A. Gurtov, Security in software defined networks: A survey, IEEE Commun. Surv. Tutor. 17 (4) (2015) 2317–2346. http://dx.doi.org/10.1109/COMST.2015.2474118.

[96] D. Kreutz, F. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, S. Uhlig, Software-defined networking: A comprehensive survey, Proc. IEEE 103 (1) (2015) 14–76. http://dx.doi.org/10.1109/JPROC.2014.2371999.

[97] Y. Li, M. Chen, Software-defined network function virtualization: A survey, IEEE Access 3 (2015) 2542–2553. http://dx.doi.org/10.1109/ACCESS.2015.2499271.

[98] M. Chen, Y. Qian, S. Mao, W. Tang, X. Yang, Software-defined mobile networks security, Mob. Netw. Appl. (2016) 1–15. http://dx.doi.org/10.1007/s11036-015-0665-5.

[99] I. Petri, O.F. Rana, Y. Rezgui, G.C. Silaghi, Trust modelling and analysis in peer-to-peer clouds, Int. J. Cloud Comput. 1 (2–3) (2012) 221–239. http://dx.doi.org/10.1504/IJCC.2012.046714.

[100] S. Chen, G. Wang, W. Jia, A trust model using implicit call behavioral graph for mobile cloud computing, in: G. Wang, I. Ray, D. Feng, M. Rajarajan (Eds.), Cyberspace Safety and Security, in: Lecture Notes in Computer Science, vol. 8300, Springer International Publishing, 2013, pp. 387–402. http://dx.doi.org/10.1007/978-3-319-03584-0_29.

[101] M. Hussain, B. Almourad, Trust in mobile cloud computing with lte-based deployment, in: Proceedings of the IEEE 11th Intl. Conf. Ubiquitous Intelligence and Computing, and IEEE 11th Intl. Conf. on Autonomic and Trusted Computing, and IEEE 14th Intl. Conf. on Scalable Computing and Communications and Its Associated Workshops (UTC-ATC-ScalCom), 2014, pp. 643–648. http://dx.doi.org/10.1109/UIC-ATC-ScalCom.2014.52.

[102] G. Shang-Fu, Z. Jian-Lei, A survey of reputation and trust mechanism in peer-to-peer network, in: Proceedings of the 2012 International Conference on Industrial Control and Electronics Engineering, ICICEE, 2012, pp. 116–119. http://dx.doi.org/10.1109/ICICEE.2012.39.

[103] Z. Yan, P. Zhang, A.V. Vasilakos, A survey on trust management for Internet of things, J. Netw. Comput. Appl. 42 (2014) 120–134. http://dx.doi.org/10.1016/j.jnca.2014.01.014.

[104] F. Corradini, F.D. Angelis, F. Ippoliti, F. Marcantoni, A survey of trust management models for cloud computing, in: Proceedings of the 5th International Conference on Cloud Computing and Services Science (CLOSER), 2015, pp. 155–161.

[105] J. Kantert, S. Edenhofer, S. Tomforde, C. Muller-Schloer, Representation of trust and reputation in self-managed computing systems, in: Proceedings of the IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM'15), 2015, pp. 1827–1834. http://dx.doi.org/10.1109/CIT/IUCC/DASC/PICOM.2015.273.

[106] M. Figueroa, K. Uttecht, J. Rosenberg, A sound approach to security in mobile and cloud-oriented environments, in: Proceedings of the IEEE International Symposium on Technologies for Homeland Security (HST), 2015, pp. 1–7. http://dx.doi.org/10.1109/THS.2015.7225266.

[107] N. Bennani, K. Boukadi, C. Ghedira-Guegan, A trust management solution in the context of hybrid clouds, in: Proceedings of the IEEE 23rd International WETICE Conference (WETICE), 2014, pp. 339–344. http://dx.doi.org/10.1109/WETICE.2014.76.

[108] A. Mtibaa, K. Harras, H. Alnuweiri, Friend or foe? detecting and isolating malicious nodes in mobile edge computing platforms, in: IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom'15), 2015, pp. 42–49. http://dx.doi.org/10.1109/CloudCom.2015.40.

[109] K. Gai, M. Qiu, L. Tao, Y. Zhu, Intrusion detection techniques for mobile cloud computing in heterogeneous 5G, Secur. Commun. Netw. 9 (16) (2016) 3049–3058.

[110] Y. Shi, S. Abhilash, K. Hwang, Cloudlet mesh for securing mobile clouds from intrusions and network attacks, in: Proceedings of the 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2015, pp. 109–118. http://dx.doi.org/10.1109/MobileCloud.2015.15.

[111] S. Iqbal, M.L.M. Kiah, B. Dhaghighi, M. Hussain, S. Khan, M.K. Khan, K.-K.R. Choo, On cloud security attacks: A taxonomy and intrusion detection and prevention as a service, J. Netw. Comput. Appl. 74 (2016) 98–120. http://dx.doi.org/10.1016/j.jnca.2016.08.016.

[112] N. Pitropakis, C. Lambrinoudakis, D. Geneiatakis, Till all are one: Towards a unified cloud IDS, in: S. Fischer-Hbner, C. Lambrinoudakis, J. Lpez (Eds.), Trust, Privacy and Security in Digital Business, in: Lecture Notes in Computer Science, vol. 9264, Springer International Publishing, 2015, pp. 136–149. http://dx.doi.org/10.1007/978-3-319-22906-5_11.

[113] W. Luo, L. Xu, Z. Zhan, Q. Zheng, S. Xu, Federated cloud security architecture for secure and agile clouds, in: K.J. Han, B.-Y. Choi, S. Song (Eds.), High Performance Cloud Auditing and Applications, Springer, New York, 2014, pp. 169–188. http://dx.doi.org/10.1007/978-1-4614-3296-8_7.

[114] C.O. Encina, E.B. Fernandez, A.R. Monge, Threat analysis and misuse patterns of federated inter-cloud systems, in: Proceedings of the 19th European Conference on Pattern Languages of Programs (EuroPLoP), 2014, pp. 13:1–13:8. http://dx.doi.org/10.1145/2721956.2721986.

[115] K. Ravichandran, A. Gavrilovska, S. Pande, PiMiCo: Privacy preservation via migration in collaborative mobile clouds, in: Proceedings of the 48th Hawaii International Conference on System Sciences (HICSS), 2015, pp. 5341–5351. http://dx.doi.org/10.1109/HICSS.2015.628.

[116] H. Zhang, N. Yu, Y. Wen, Mobile cloud computing based privacy protection in location-based information survey applications, Secur. Commun. Netw. 8 (6) (2015) 1006–1025. http://dx.doi.org/10.1002/sec.1055.

[117] X. Huang, R. Yu, J. Kang, N. Wang, S. Maharjan, Y. Zhang, Software defined networking with pseudonym systems for secure vehicular clouds, IEEE Access 4 (2016) 3522–3534. http://dx.doi.org/10.1109/ACCESS.2016.2560902.

[118] B. Cruz Zapata, J.L. Fernandez-Aleman, A. Toval, Trust modelling and analysis in peer-to-peer clouds, Comput. Sci. Inf. Syst. 12 (1) (2015) 161–184. http://dx.doi.org/10.2298/CSIS140205086C.

[119] R. Di Pietro, F. Lombardi, Security for Cloud Computing, Artec House, Boston, ISBN: 978-1-60807-989-6, 2015.

[120] Y. Sun, J. Zhang, Y. Xiong, G. Zhu, Data security and privacy in cloud computing, Int. J. Distrib. Sens. Netw. 2014 (2014) 1–9. http://dx.doi.org/10.1155/2014/190903, (Article ID 190903).

[121] S. Seneviratne, A. Seneviratne, P. Mohapatra, Personal cloudlets for privacy and resource efficiency in mobile in-app advertising, in: Proceedings of the 1st International Workshop on Mobile Cloud Computing and Networking (MobileCloud), 2013, pp. 33–40. http://dx.doi.org/10.1145/2492348.2492356.

[122] A. Page, O. Kocabas, S. Ames, M. Venkitasubramaniam, T. Soyata, Cloud-based secure health monitoring: Optimizing fully-homomorphic encryption for streaming algorithms, in: Proceedings of the 2014 Globecom Workshops, 2014, pp. 48–52. http://dx.doi.org/10.1109/GLOCOMW.2014.7063384.

[123] J. Abdo, J. Demerjian, H. Chaouchi, T. Atechian, C. Bassil, Privacy using mobile cloud computing, in: Proceedings of the 5th International Conference on Digital Information and Communication Technology and its Applications (DICTAP), 2015, 178–182. http://dx.doi.org/10.1109/DICTAP.2015.7113194.

[124] F. Lombardi, R. Di Pietro, Virtualization and cloud security: benefits, caveats, and future developments, in: Cloud Computing: Challenges, Limitations and R&D Solutions, Springer International Publishing, 2014, pp. 237–255. http://dx.doi.org/10.1007/978-3-319-10530-7_10.

[125] S. Berger, R. Cáceres, K.A. Goldman, R. Perez, R. Sailer, L. van Doorn, vTPM: Virtualizing the trusted platform module, in: Proceedings of the 15th Conference on USENIX Security Symposium (USENIX-SS'06), 2006.

[126] W. Ma, X. Li, Y. Shi, Y. Guo, TVMCM: A trusted VM clone model in cloud computing, in: 6th International Conference on New Trends in Information Science and Service Science and Data Mining (ISSDM'12), 2012, pp. 607–611.

[127] M. Aiash, G. Mapp, O. Gemikonakli, Secure live virtual machines migration: Issues and solutions, in: 28th International Conference on Advanced Information Networking and Applications Workshops (WAINA'14), 2014, PP. 160–165. http://dx.doi.org/10.1109/WAINA.2014.35.

[128] L. Jacquin, A. Lioy, D.R. Lopez, A.L. Shaw, T. Su, The trust problem in modern network infrastructures, in: Cyber Security and Privacy: 4th Cyber Security and Privacy Innovation Forum, Springer International Publishing, 2015, pp. 116–127. http://dx.doi.org/10.1007/978-3-319-25360-2_10.

[129] Z. Hao, Y. Tang, Y. Zhang, E. Novak, N. Carter, Q. Li, SMOC: A secure mobile cloud computing platform, in: Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM), 2015, pp. 2668–2676. http://dx.doi.org/10.1109/INFOCOM.2015.7218658.

[130] N.M. Dhanya, G. Kousalya, Adaptive and secure application partitioning for offloading in mobile cloud computing, in: J.H. Abawajy, S. Mukherjea, S.M. Thampi, A. Ruiz-Martnez (Eds.), Security in Computing and Communications, in: Communications in Computer and Information Science, vol. 536, Springer International Publishing, 2015, pp. 45–53. http://dx.doi.org/10.1007/978-3-319-22915-7_5.

[131] C. Borcea, X. Ding, N. Gehani, R. Curtmola, M. Khan, H. Debnath, Avatar: Mobile distributed computing in the cloud, in: Proceedings of the 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2015, pp. 151–156. http://dx.doi.org/10.1109/MobileCloud.2015.22.

[132] A. Pichan, M. Lazarescu, S.T. Soh, Cloud forensics: Technical challenges, solutions and comparative analysis, Digit. Invest. 13 (2015) 38–57. http://dx.doi.org/10.1016/j.diin.2015.03.002.

**Rodrigo Roman** is a security researcher working at the University of Malaga (Spain), where he obtained his Ph.D. and M.Sc. degrees in Computer Engineering and Computer Science, respectively, in 2008 and 2003. Previously, he worked for the Institute of Infocomm Research (I2R) in Singapore in the areas of sensor network security and cloud security. Pursuing to make security simple and usable, his research is focused on the development of protection mechanisms for the Internet of Things and related paradigms.

**Javier Lopez** is Full Professor in the Computer Science Department at the University of Malaga, and Head of the NICS Lab. His research activities are mainly focused on information security, future Internet security, and critical infrastructure protection, and has lead several international research projects in those areas. Prof. Lopez is Co-Editor in Chief of IJIS journal and the Spanish representative in the IFIP TC-11 on Security and Privacy Protection in Information Systems.

**Masahiro Mambo** received a B.Eng. degree from Kanazawa University, Japan, in 1988 and M.S. Eng. and Dr.Eng. degrees in electronic engineering from Tokyo Institute of Technology, Japan in 1990 and 1993, respectively. After working at Japan Advanced Institute of Science and Technology, JAIST, Tohoku University and University of Tsukuba, he joined Kanazawa University in 2011. He is currently a professor of Faculty of Electrical and Computer Engineering, Institute of Science and Engineering. His research interests include information security, software protection and privacy protection.