Survey on Identity and Access Management for Internet of Things

BRUNO CREMONEZI (✉ bruno.cremonezi@hotmail.com)

Universidade Federal do Parana

Alex Vieira

Universidade Federal de Juiz de Fora

José Augusto Nacif

Universidade Federal de Vicosa

Michele Nogueira

Universidade Federal do Parana

Research

Keywords: Identity Management, Internet of Things, Authentication, Authorization, Access Control

Posted Date: September 8th, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-66793/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

RESEARCH

Survey on Identity and Access Management for Internet of Things

Bruno Cremonezi^{1*}, Alex Vieira², José Augusto Nacif³ and Michele Nogueira¹

*Correspondence:

cremonezi@ufpr.com

¹Universidade Federal do Paraná,

Full list of author information is available at the end of the article

[†]Equal contributor

Abstract

The Internet of Things (IoT) encompasses a large number of connected devices, generating and sharing different types of data among themselves. These data enable the creation of society-changing applications, such as health monitoring and autonomous vehicles. Under this context, protecting the access of these connected devices and their data is critical to IoT applications' success, since a single data breach can incur into a cascade effect, which leads to devastating consequences. Identity and Access Management (IAM) systems provide mechanisms to identify individuals at the network and determine their access privileges, thus avoiding inappropriate access. However, the current IAMs system struggles to provide the scale or manage the complex relationships that IoT brings. In this work, we present a comprehensive state-of-the-art survey of IAMs and the main concepts and challenges when applied to IoT. First, we overview the IoT technology, giving its essential characteristics and communication architectures and its main applications. Then, we present IAMs state-of-the-art and the main concepts, existing architectures, and challenges. Finally, we focus on current IAMs that aims to tackle the IoT complexity, along with an in-depth analysis of their proposals and future directions in the field.

Keywords: Identity Management; Internet of Things; Authentication; Authorization; Access Control

1 Introduction

The paradigm of the Internet of Things (IoT) has gained ground in the scenario of wireless communications [1]. The idea of having millions of objects interconnected under the control of human beings has paved the way for a diverse set of applications that promise to solve problems that are currently challenging in society, such as autonomous driving, assisted living, and e-health. These are only a few examples of IoT-based applications [2] among a myriad of others. Today, this hyper-connected digital world is quickly becoming a reality. The entire network ecosystem is conditioned to have a massive explosion in the number of connected objects worldwide. For instance, there are more than 25 billion devices connected around the world, and the forecast for 2025 is that this number surpasses the mark of 75 billion [3].

In this scenario, security and accessibility are must-have foundations [4]. The damage that a malicious user achieves in compromising a connected vehicle, a copier connected to a corporate network, or even a medical device is vast since IoT devices can act as a “back door” to launch significant attacks putting a network's infrastructure and people's life at risk [5].

The security of IoT devices passes through managing their identities. Access control is critical to the success of the IoT since most of the information contained in

the IoT environment may be personal or sensitive [6]. In its definition, an Identity and Access Management system (IAM) plays a fundamental role in assisting the network infrastructure by providing information about user profiles, service features, and access policies. Today, IAM shows itself as a consistent way to enforce business and security policies, regardless of network entry point by the users. It is also a way to manage user's identities and identify users and devices to control their access to network resources [7]. In short, the core of an IAM comprises policies defining which devices and users are allowed on the network and what a user can accomplish, depending on device type, location, and other factors [8].

To keep pace in this hyper-connected context proposed by IoT, an IAM system must manage more identities than never before. Instead of managing users' identities and a few devices, the IAM on the IoT era must address all the entities on the network, from devices and systems to individual network users. Thus, an IAM infrastructure must be designed to scale to the number of devices, with each device having a multi-faceted relationship with other network entities. A device needs to communicate with other devices, end-users, and applications, meaning that at every second, there is a considerable number of different relationships that have to be controlled, managed, and secured. Since this context is dealing with a massive number of devices spread globally, we need to quickly provide new identities to devices with the correct access rights - but we also need to be able to update and revoke them just as effectively [9].

In this work, we present a comprehensive survey of IAMs for the IoT. We analyze the main concepts and challenges of this promising area, including new IoT applications and the particular challenges they introduce to the current IAM systems. We also point to future directions considering recent work found in the literature. To summarize, we provide the following contributions:

- We explore the fundamentals of IoT and provide an overview of the representative applications, main characteristics, and architectures;
- We provide a comprehensive insight into what comprehend a digital identity;
- We give an overview of IAM and its architecture, including operations, models, and challenges. We also explain how IoT is influencing on the future of IAMs;
- We delve into an analysis of IAMs proposals for IoT.
- We give a perspective of the future of IAMs for IoT and a broader perspective of the area considering the recent advancements in the field.

The remainder of this paper is organized as follows: In Section 2, we overview the main characteristics of IoT, together with typical applications and communication architectures. In Section 3, we present the definition and concepts of digital identities, together with an explanation of its lifecycle. In Section 4, we give an overview of IAM, presenting the definition, evolution, and classification of identity management models. We analyze the requirements for IAM for IoT and several proposals designed for it in Section 5. In Section 6, we give a future perspective to IAM challenges in the IoT. Finally, in Section 7, we present the final discussions, as well as a broader perspective of the area.

2 Brief Review on Internet of Things

The term **Internet of Things** (IoT) has been around for quite a few years and has attracted the attention of researchers worldwide. IoT refers to a type of network to connect “everything” (for example, devices, applications and services, human users, data, communication endpoints, and locations) through some communication technology, such as Bluetooth, Zigbee, WiFi, Celular, and LoRaWAN [10]. In this network, “everything” is uniquely identifiable and addressable, meaning that they can be individually targeted and found at the network. Therefore, we define that IoT is a network that allows persons, real-world objects (devices, communication endpoints) and virtual entities (applications, services) to interact with each other over the Internet by their unique identifier to achieve some goal [11].

In the following, we further characterize and describe IoT in terms of (i) applications, (ii) its main characteristics and, (iii) and the main architectures used in IoT systems.

2.1 IoT Applications

IoT has gained popularity in recent days due to the capacity to interconnecting “everything” and the potential to offer creative applications that deliver substantial benefits to society. It is already possible to find applications proposals of almost everything serving different purposes, from simple home control to autonomous vehicles. In this section, we present and discuss three examples, among a variety of IoT based applications: Smart Home, Healthcare, and Connected Vehicles. These applications illustrate the particularities of IoT and, at some level, are already deployed in today’s world or are currently planned to appear in the next years.

Smart Home: Initially, the term smart home was used to define an environmental control system, such as lighting and heating control. However, with the introduction of IoT, this term expands to any device within the house, which has now included smart TVs, smart thermostats, smart security cameras, and other similar devices. The main idea of a smart home is to have these devices operating together to provide homeowners security, comfort, and convenience [12]. Smart homes demonstrate the potential for the development of a wide range of applications due to the variety of house objects. Burglary prevention, for example, allows homeowners to enhance their house security, performing monitoring even when they are away. A smart motion sensor within the house captures any suspicious movement and sends a notification directly to the homeowner’s smartphone. Once properly notified, homeowners connect their smartphones at the house’s smart security camera system to have real-time video streams of their home, providing evidence to confirm the suspicious activity. Regarding comfort and convenience, a Smart Gardening application takes advantage of smart sensors to automate whether to increase or decrease water supply or even collect data on incoming weather patterns to determine the most suitable course of action [13].

Healthcare IoT: The current healthcare system emphasizes physical visits to hospitals with the procedures for patient monitoring, care, management, and supervision manually executed by nursing staff. These constant physical visits may represent a bottleneck that burdens the nursing staff, leading to tragic errors in practice. With the arrival of IoT technology, healthcare providers use IoT medical devices to reduce costs and improve the efficiency of patient care [14]. Given

that healthcare is a vast ecosystem, the application of IoT in this scenario seems to be endless. Diabetes treatment, for example, can be done through IoT devices. A patient wears a continuous glucose monitor, which monitors the patient's blood glucose levels for several days and takes readings at regular intervals. Those readings feed an insulin delivery device that automatically adjusts the amount of insulin injected into the patient's body, helping them keep blood glucose within the safe range, preventing human errors such as extremely high or low doses [15].

Connected Vehicles: Connected vehicle refers to applications, services, and technologies that connect a vehicle to its surroundings. Thus, each vehicle has several devices connected to other internal or external devices, networks, applications, and services [16]. This connectivity allows the vehicle to share a high volume of transportation-related data, enabling plenty of new applications, such as traffic management, urban and suburban mobility, accident prevention, motorist advisories and warnings, and even autonomous self-driving vehicles [17, 18].

2.2 IoT Characteristics

IoT creates opportunities for a wide range of applications. Those applications may have their specific domain requirements and characteristics. However, some of the IoT characteristics are generic enough that arise in a variety of domains [19, 20]. Next, we discuss the most significant IoT characteristics described by Patel et al [19].

Scalability: IoT devices generate data that feeds several applications. For example, several connected vehicles track the road information level and deliver it to a government server. This information feeds an application that allows traffic efficiency visualization, helping traffic engineers plan more safe and efficient traffic flow. Once the number of devices connected composing the IoT increases each day, the amount of data IoT devices generate tends to grow. Then, an IoT architecture must be capable of handling this massive expansion of devices and data [20].

Heterogeneity: IoT connects different devices with different capabilities, complexity, and vendors. Devices in IoT rely on various hardware platforms and networks, interacting with other devices or service platforms through different networks. Hence, IoT architecture must support network connectivity between heterogeneous devices and networks [20].

Connectivity: From an idealistic viewpoint, IoT comprises of a global scale, to thousands or even millions of devices simultaneously connected to the Internet. However, the Internet is composed of heterogeneous networks and communications technologies. Emerging IoT applications, such as healthcare and connected vehicles, demand ubiquitously available Internet connectivity. Thus, to enable this connectivity, IoT must consider a wide range of protocols and communication technologies, such as LPWANS, Bluetooth, Cellular (3G/4G/5G), Zigbee, 802.11, RFID, among others [19, 20].

Dynamicity: IoT devices are exposed to rapidly changing surroundings, experiencing different connections, and disconnections throughout their lifetime. In general, many IoT devices are low energy and lightweight, meaning that most of them rely on batteries as their primary power resource. Thus, if a device's battery dies, the network experiences a disconnection. Furthermore, mobile devices are fundamental

to creating several IoT applications, such as self-driving vehicles. These devices are subject to rapidly network changing, experiencing connections, and disconnections with several devices while they are on movement due to their mobility. Thus, the IoT device must dynamically adapt in response to constant changes, meaning that it needs to reconfigure to avoid disruptions on applications [21, 19], ensuring seamless connectivity and management without human interaction.

Privacy and Safety: In several applications, such as self-driving vehicles and personal glucose monitor, the IoT devices have a role in generating, processing, and sharing private and safety-critical data. However, most of them present a lack of battery and computational power. Consequently, the devices spend most of their resources executing their primary function, leaving only a few resources for privacy and safety functions. This lack of resources turns the act of establishing defense techniques in the context of attacks in IoT much more complicated than conventional information systems [22], once those traditional security methods tend to be expensive for IoT in terms of energy consumption and processing overhead [23]. Besides, most traditional security demands a highly centralized architecture, which may not be suited for IoT due to the difficulty of scale and the mobility pattern of several IoT devices [23].

2.3 IoT Architectures

There is no single architecture of IoT that is widely accepted by the scientific community. Hence, several works focus on proposing new ones. However, the most accepted architectures in the literature are the *3-layer architecture* and *5-layer architecture* [24].

3-Layer Architecture: Introduced in the early stages of IoT research, this architecture defines three layers: the perception, network, and application layers. The perception layer is the physical layer, whose purpose is to collect data, to identify the interacted entities, and to perform actions using specific equipment such as sensors, actuators, and readers [25]. The network layer is the core layer of the IoT, and it is responsible for transmitting the gathered information by the perception layer with the use of wire/wireless networks and the Internet. Finally, the application layer is responsible for delivering application-specific services to the user. It defines various applications for IoT deployment [25].

5-Layer Architecture: Several works present new proposals for IoT architectures [26]. One of the most accepted architectures is the five-layer, which expands the traditional three-layer architecture by including two new layers: processing and business layers. The five layers are perception, transport, processing, application, and business layer. The role of the perception and application layer remains the same as the same that they have on the three-layer architecture. The transport layer is responsible for transferring the collected data from the perception layer through wired/wireless networks. However, instead of transferring data to the application layer, they are sent to the processing layer. The processing layer analyzes, stores, and processes the data that comes from the transport layer and can also manage or/and provide a diverse set of services to the lower layers. It employs many technologies, such as databases, cloud/fog computing, and big data processing modules. Finally, the business layer has the purpose of managing applications, business, and profits models of IoT [27, 25].

3 Identities for the Internet of Things

IoT introduces the concept of a world where everything becomes interconnected, allowing “things” to communicate. This communication creates opportunities for a wide range of applications; however, to be securely connected, “everything” first needs to be identified. Therefore, we can define identity as the digital representation of the information known about a specific person, device, or service into IoT. Since IoT covers this wide variety of “things”, we define humans, devices, and services as subjects of identities. Thus, identity is a digital representation of a subject made by itself or by another subject [28]. We can use the identity information for different purposes, ranging from allowing subject to prove his, her, or its claim to identity until establishing permissions to enable interaction with other subjects [7, 29, 30]. In this section, we first overview these identities, presenting the basics of the notion of identity and its components. Second, we discuss the life cycle of these identities, describing what happens since the creation until the revocation of one subject’s identity.

3.1 Components of an Identity

Identity is a digital representation of the subject, where this representation is done through a set of claims [28]. These claims are referred to as identity attributes and present the characteristic elements of their subject. Each identity is exclusively associated with a subject; however, the opposite is not valid. A subject can have more than one mapped identity, where each identity encompasses valued attributes within an application context. We call these multiple identities partial identities of the subject and denote the completed identity as the set of combinations of these “partial identities” that are used for specific application contexts [30]. Furthermore, identities can be permanent or temporary, depending upon the application’s context [31]. For example, a subject may have a permanent identity and another as a company’s interim accountant. Figure 1 shows an example of a subject (Alice) with more than one identity, each representing her in a different context. As an employee, Alice’s identity consists of a series of attributes that indicate her role inside a company, describing her name, job title, job category, and any other attribute that company needs. As a user of music streaming applications, Alice is represented by other types of attributes, such as name, gender, and favorite music genre.

Figure 1 Relationship among subjects, their identities and attributes.

According to the ITU-T recommendation (Y.2720) [32], an identity is composed of three different components: **identifiers**, **attributes** and **credentials** [32]. An

identifier is a series of digits, characters, symbols, or any other form of data used to index one identity in some context. **Attributes** are pieces of data bounded to an identity that specifies a characteristic of the subject owner of that identity, such as condition, quality, or other information associated with that subject. **Credentials** are a set of data that can be used to the subject to claim its identity. Therefore, credentials link the subject with their identity in a process called **authentication**, which details we discuss in Section 4.

3.2 Identity Lifecycle

We define an application as any service or group of services available for registered subjects registered. These applications can utilize the Internet or other network hardware infrastructure to perform useful functions, such as data sharing. When a subject is registered in some application, we refer to them as users of that application. A lifecycle is the definition of phases to identify an object status in a period. Therefore, an identity lifecycle determines the status of a user's identity within an application context. In Figure 2, identities have a generic lifecycle framework that is applicable regardless of the application, and it is composed of five phases: provision, propagation, usage, maintenance, and de-provisioning. We describe all phases in the following subsections, defining their role in the identity life cycle.

Each application context may have its identity lifecycle, and planning each phase is essential to building an identity architecture. Briefly, an identity starts out being provisioned or created for a subject. After created, the subject becomes a user of that application, and their identity is propagated by any application that utilizes this user's identity information (details in Section 4). Once propagated, applications utilize the identity, and occasionally some identity changes, such as credentials changing or attributes addition, may occur. In these cases, identity updates force an identity to propagate again. Finally, when this identity serves its purpose and is no longer needed, it is disproved or destroyed [33].

Identity Provision: This term represents the creation of an identity for a subject, a step before becoming a user of the application. Therefore, Identify Provision creates a unique identifier, credentials, and the record of the subject's attributes. These attributes can be, for example, location, email, and specific attributes for an application context. Some applications require an explicit identity provision for their users. In those cases, which are usually persons, the user registers himself into an application sending the attributes and the identity proof to be used as credentials. The application checks the authenticity, validity, and accuracy of these attributes before establishing a link between subject and identity [34]. However, there are cases where the identity of the subject is not explicitly established. In those cases,

it is possible to construct a digital identity of the users, based on the collection of various network attributes used in various contexts. Most of the time, the subject is not aware of this implicit way of the construction of digital identity [35].

Identity Propagation: Some applications require that pieces of identity propagate to other systems. This replication objective is simple: applications may replicate the identity for better performance, cost, or simple failure defense system. More complex applications may require a unified identity directory where an identity created by some application may be used in another application. Ideally, a propagation must occur after each change in some identity, and the propagation must occur in a reliable way to avoid the problems of safety and consistency [33].

Identity Usage: This is the straightforward phase of the identity lifecycle. During this phase, several applications and users use this identity to perform identity verification, which can determine if a user's identity is legitimate and access control operations, which allow a user the access to perform actions over resources [33].

Identity Maintenance: In a general approach, identities are not static, and attributes and credentials may experience several changes during the identity lifecycle. For example, the base user characteristic may change over time. Its identity must follow up this change or, as another example, the application must support new business opportunities, requiring a complete change on the identity by adding new attributes. Independent of the factor that motivates this change, after completion, the results must be propagated on all affected applications [36]. Due to the dynamic nature of IoT, this is one of the most costly phases of the identity life cycle.

Identity Deprovision: Removing identities at the end of its lifecycle is just as crucial as providing those identities. Deprovisioning is the process which enables the application to know about which users are no longer valid [37]. The most straightforward approach to Identity Deprovision comes with the complete removal of a user's identity. However, deleting an identity means removing all tied information, which might still be necessary for auditing. Thus, applications opt to disable instead of deleting identities. In those cases, the applications revoke the identity's credential, meaning that the identity still exists but is no longer linked to a user and does not have any access rights associated with it. Hence, the application minimizes the loss of information and stays secure; however, the cost to maintain the information of disabled identities may be encumbered. For this reason, some applications utilize a hybrid approach. They implement a mechanism that disables an identity first but deletes it only after a time interval. Therefore, this approach aims to combine the benefits of both previously mentioned approaches; however, it keeps the drawback of the first one after the deletion [38].

4 The State-of-the-art in Identity and Access Management

Identity and Access Management (IAM) is a system for managing the life cycle of users' digital identities [39]. It encompasses provisioning and de-provisioning identities, identity authentication, and authorization to a user to access services. In a nutshell, the main goal of identity and access management is to ensure that only authenticated users have access to specific services [40]. Figure 3 helps to give a complete overview of the identity and access management topics that we will address in this survey. We will start providing a brief history of this system, following by

the authentication and authorization methods, and finally, the identity and access management models most commons today.

4.1 A Brief History of Identity and Access Management

IAM has started as a simple security solution to check the identity of persons when accessing a specific application service [41]. In this beginning era of IAM, the group of users of an application is mainly composed of persons, meaning that IAMs focused on a human’s identification. When accessing an application service, a person must first register herself on it. When they complete registration, the IAM creates their identity in that service. Latter, to prove her identity, the user must memorize a username and password. This identity represents that person on the service and is used to determine what she can do. When a user wants to use that service again, she proves her identity through login, inputting her username and password [7]. At this beginning, the IAM system was exclusively attached to one specific service, and this exclusivity on each service turns out that each one should include its own isolated IAM. If a user wants to use another service, she must register himself at that service and pass through the log in again, repeating all the processes of inputting a username and password. Although sufficient to guarantee service access security, the increasing number of services may be a burden to the user. Once that, with several IAMs, the user should memorize a username and password for each service, making this task difficult and annoying over time [42].

To solve this problem, IAMs have been detached from a single service and began to serve multiple services. Hence, a user has a single identity to represent her under a wide range of services. From the user point of view, they must have to log in only one time to access various services, which makes the password memorization process much more straightforward than before [43].

Under the same IAM, a user identity may be the same, no matter the service she logs in. This feature enables identities to carry valuable information such as preferences, location, and history of activities. As a result, services have gained the ability to verify the identity of users and make personalized decisions based on their behavior [44]. However, not only the user experience becomes enhanced, but service security also benefits since IAMs can instantly spot fraudsters based on their past activities. By detecting anomalies, IAM can, for instance, identify scammers who are impersonating legitimate users [45].

With the deployment of IoT, IAM and its concepts are once more put into proof. In this context, the term “user” is not just persons interacting with services; it is

a wide variety of things, ranging from objects to individuals communicating and exchanging data among themselves. Therefore, having an identity is very important to allow IoT devices to determine “who or what it is communicating” and if “she or it has rights to communicate” [46].

4.2 Authentication, Authorization, and Auditing

IAM must integrate policies and technologies to enable managing user information and to control their access to online services. Since identities contain attributes of a user (both humans and non-humans), IAM must provide access for those users, while preserving confidential personal and business information from unauthorized user [47].

In a simplified way, Identity Management is about managing identifiers and attributes related to identity, while Access Management is about evaluating these attributes based on policies and making decisions. Therefore, considering an IAM, three main operations are enumerated: **Authentication**, **Authorization** and **Auditing** [48]. Figure 4 illustrates a coherent picture of these three operations and their interactions, showing a situation in which a user wants to access an available network service. However, to access this service, the user must go through the authentication and authorization operations. The auditing operation, in turn, supervises all this process, creating a log for each output of the authentication and authorization operations.

In Figure 4, the first barrier in controlling access to network services consists of verifying that a user who wants to access service is who she claims to be. The **authentication** operation main idea is that each user has some unique information that sets it or her apart from other users. During the authentication, a user provides credentials for the claimed identity, ideally, knew only by herself. As a result, IAM matches the identity’s credential to the credential offered by the user and concludes if that user is the rightful owner of the chosen identity [48].

Once the system finalizes user authentication, her access still depends on the rights that the user has. Therefore, the **authorization** operation is a process of granting or denying the user access to some services based on a set of rules. In most systems, not all users should have the same rights to perform specific actions. Therefore, the access control model is crucial to protect specific services from unauthorized access. An access control model determines which user and under which conditions/policies can access a service [49].

Auditing means monitoring user activity, recording every action done, since the authentication until the events that follow granted user authorization. With Auditing, the system keeps track of identities activity, such as authentication results and which services have been accessed. Keeping track of users and their activities serve many purposes. For example, tracing back to events leading up to a security incident can prove very valuable to a forensics analysis and investigation case [50].

In a nutshell, it is essential to point out the clear distinction between authentication and authorization. The recognition of the identity of a user is the responsibility of the authentication operation. Authorization assumes that authentication has done correctly and applies the Access Control model. Therefore, the effectiveness of an authorization rests on proper authentication and correctness of the Access control model. The audit concerns the post-analysis of all the requests and activities of both authentication and Authorization operations requested by a user on the system [49]. Therefore, the audit is useful to discourage attempting violations, analyze user behavior, and to track possible flaws in the access control model.

4.2.1 Authentication Methods

The authentication describes the process of verifying a user's ownership over an identity. However, such proof comes in different ways, and it is associated with the identity credentials. There are various types of credentials, usually classified as: *knowledge*, *possession*, *inherence* and *context-aware* factor [51, 52]. Except for the inherence factor, all authentication methods are valid to both human and non-human users.

Knowledge factor credential describes a piece of information that the user knows. Usually referred to as “any information of something that it knows”, this method is widely used today. For persons, the standard user name and password authentication process is a typical example of this category of credential type. Despite its popularity, this authentication method has the drawback of depending on user memorization capacity, meaning that the user should never forget the passwords, and once that without them, the user loses access to her identity. For IoT devices, the device stores the password, which arises a series of security concerns, for example, the challenges for a human user when managing a large number of device usernames and passwords, and the complexity of securing the passwords stored on the devices.

Possession factor credential describes as “something that the user must have in their possession” to progress with the authentication. When a user is a person, the most common possession credential may include a one-time password (OTP) generator, ID card, or a smartphone. If we take into account that possession factor credentials can be lost, this type of credential faces the same challenges and problems of the knowledge factor credential. For IoT devices, secrets stored in the device prove the possession factor credential. In this case, a device store a piece of information that the system recognizes as a reliable proof of identity. For example, an IoT device can establish a symmetric key cipher with another party, meaning that both need to have the same key acting as a proof of identity.

Inherence factor credential are exclusive to human users, and constitutes of biological characteristics, such as voice and fingerprint, for the authentication process. Referred as “something that user is”, this factor has the benefit of that cannot

be lost; however, it can be temporarily unavailable due to, for example, damaged fingerprints or even common throat problems like hoarseness. In general, most of the biometric constitutes of public information, turning this factor prone to be replicated by malicious users, like a fake fingerprint or a voice record. As a result, this kind of credential is not secure to use alone, and several works point out that must be used in conjunction with other factors.

Context-aware factor credential is often suggested as a complementary credential that increases the robustness of the authentication method [53]. For human users, it consists, for example, of verifying the user location using GPS devices combined with the time, which enables an accurate confirmation of the identity of the user. For devices, the system confirms the identity using behavior or characteristics, such as geographic, and communication technology [53].

4.2.2 Authorization Methods

The success of an authorization method is related to the access control model, which determines a set of requirements that a user must meet before access a service. Diverse access control models have been proposed for use in IoT systems, such as Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), and Capability-Based Access Control (Cap-BAC). Each one of them has its upsides and downsides.

Discretionary Access control (DAC) [54] is an access control mechanism that was initially developed for operating systems, and later transposed for the IoT context. In DAC, an owner of an IoT device specifies which users can access it and defines some access rules, such as which operations are valid, and which hours other users can access. Several approaches have been proposed to implement the DAC: access matrix, authorization table, and access control list (ACL). In general, for a single device, this approach gives the owner full control, identifying who can access it and under which conditions the operations are accessible to them. However, if the user is the owner of a high number of devices, the lack of a centralized administration can turn the design of access conditions complex and the auditing process complicated [55].

Mandatory Access Control (MAC) [56] is an access control model based on the classification of all entities in the IAM. In this model, each user (both human and non-human) and service has a security label, which reflects the sensitivity of the information that they can access or generate. The security label reflects the user's trustworthiness not to disclose sensitive information. To function correctly, MAC models put several restrictions to limit the label changes, allowing only a limited set of human users to modify object security labels. For this reason, MAC models are difficult and expensive to implement and maintain, particularly in dynamic scenarios that require more flexibility, for example, in a patient's emergency the healthcare application must lower the security of the user data to provide a faster response. However, if the application utilizes the MAC, only a few persons can change the security label, which can put the person's life at risk, since the healthcare professionals cannot receive this information in time [55].

Role-Based Access Control (RBAC) [57] is one of the most used access control model. Each user of the applications has a role, and this role determines which

services and operations she can access. Thus, users are assigned to roles and inherit the permissions assigned to that role. Roles can also be organized in a role hierarchy, defining that a role can inheritance the permission of other roles. In general, this model provides effective access management, but it defines only predefined static roles. The definition of these roles are highly dependent on a centralized entity, and depending on the complexity of the application, the number of roles can rapidly increase, turning this method unfeasible and potentially cumbersome to the IAM [58].

Attribute-Based Access Control (ABAC) [59] is similar to RBAC, however it is more flexible. Instead of defining a role, a set of policies tests attribute conditions, allowing or not access to some service. This strategy provides a fine-grained access control model. However, there are questions around the ideal number of policies and their evaluation [58]. These questions become more complicated when they assume that these attributes can be given from multiple resources (for example, the access depends on two user's identities) and can change over time, leading to safety and consistency problems [36].

Capability-Based Access Control (Cap-BAC) [60] is an access control model based on tokens that contain rights granted to the user that holds it [61]. Thus, during the authentication process, tokens are created by IAM and sent to the user. Each token directly identifies target services, the user to which the system has granted the rights, and operations allowed. Consequently, the user needs to show this token to the service before requesting an operation. The main disadvantage is the need for IAM to create and maintain all tokens, which also involves determining the policies of tokens creation[62].

4.3 Classification of Identity and Access Management

Identity and Access Management is continuously reshaping to follow technology changes. In the first generation of IAMs, there was no separation between entities offering services or identity information; therefore, each service manages the identity of its users in total isolation [63]. However, across time, this model has ended up turning outdated to some applications, and new models have emerged. In this survey, we define that an IAM falls off five basic models: isolated, centralized, federated, user-centric, and self-sovereign model. In isolated, centralized, federated, and user-centric models, the user always relies on third-party identity providers to store and share their data. On the other hand, the self-sovereign model enables a user to have all their digital identity data stored and managed by her, allowing a user to share its information with selected SPs selectively. Figure 5 illustrates this classification, showing the evolution of IAM models across time, showing evolutionary path since isolated until self-sovereign model. For each model, we present in following subsections the main features, benefits, and limitations.

The core concept of the isolated, centralized, federated, and user-centric IAMs includes three interdependent entities: User, Service Provider (SP), and Identity Provider (IdP). *user* is an entity (both human and non-human) that intends to gain access to some service. *Service Provider* is the entity that offers that service. Typically, they do not offer any method to vouch for the user's identity; instead, it

relies on the Identity Provider to do this. Finally, the *Identity Provider* is the entity that stores and vouchers for identities of users [48].

Isolated Model: This model has strongly evolved with centralized computing. Under this model, a user that wishes to access a service offered by an SP must first prove her identity. This identity has the user's attributes, and the SP manages these attributes in isolation. Therefore, the main characteristic of this model is that the SP assumes the responsibility of an IdP, managing and storing all of its users' identities and attributes [64]. Figure 6 shows an example of an isolated model. As shown, a user registers herself on two SPs in which she wants to get the services. For each SP, the user has a unique identity with her attributes, and each identity has unique credentials. Once that there is no cooperation between these two SPs, each SP assumes the role of own IdP, managing the user's identity in total isolation from another.

The simplicity of this model comes with some drawbacks. Managing identities in the local scope are straightforward, which allows simple authentication, authorization, and accounting. However, scalability becomes apparent as the number of identities grows. Once that a user has different identities on each SP registered, this model can induce reused credentials or make some of them be forgotten [65]. Besides, this model can wound the privacy of the registered user, because every service has her identity with all required attributes. Finally, due to the isolation, sharing an attribute among two or more identities change compromises the propagation of this change since, in this model, there are identities spread across different SPs.

Centralized model: This model does not confine the scope of identity to a single service, turning the network of services that create the identity responsible for the bounding. Therefore, this model introduces a central IdP, which is the identity authority, that centralizes IAM. Once establishing an identity, a user can use any SP attached to that IdP without having to engage in the authentication process explicitly. This concept - known as a Single Sign-On (SSO) - allows a user with one unique identity access to multiple services [64]. Figure 7 shows one example of the

centralized model. Two SPs agree to deliver this task to a centralized IdP instead of being their own IdP. Hence, when the user registers itself on that central IdP, she has a unique identity that grants her access to all SPs depending on that same IdP. Then, different from the isolated model, the user can have access to both SPs using the same identity. With this centralization, the user only needs to memorize one identity and its credentials instead of multiple identities and credentials. However, this centralization is a doubtful advantage, because if one identifier with the associated credential is compromised, all services that identity can access are also compromised. Furthermore, the centralized aspect of this model does not solve the problem of scalability of a large number of users or services.

Federated Model: The concept of federation represents the relationship between two or more organizations that have identity infrastructure capabilities [63]. In this model, a group of IdPs and SPs is bound together to form a federation. In this case, ruled by a set of commercial agreements and a standard technology platform, a user participating in one organization can directly access SPs at another organization. The result is that a user participating in the federation has an extension of services without the need to manage its identity in other organizations. In other words, this model allows several identity authorities to divide the power of a single one [66].

Just like the centralized model, SSO is also allowed at the federation. In this case, a user can authenticate itself a single time with a single IdP, and all IdPs of federation consider that user authenticated, enabling that user access to all SPs that are also members of the federation. In its pure form, in the identity federation, a user only needs to have one identity profiled, generally at its home organization. However, an identity ends up spanning at multiple IdPs participating in the federation. This identity spanning occurs because, as much as avoiding redundancies is the core of federation design, sometimes an IdP still needs to replicate an identity for its internal management or even for better performance or to reduce costs and risks of failures [63, 36].

Once identities can carry valuable information, there are several regulations concerning privacy protection and identity disclosure. Therefore, performing identity replication carelessly leads to security and privacy problems. Thus, it is essential to inform the purpose of identity replication. An approach to fulfill privacy and security requirements is through the use of partial identities with pseudonyms [67]. In this approach, a user's identity is replicated only to carry the essential needed original identity's attributes. The pseudonyms are new identifiers of this replicated identity, which helps to achieve something close to anonymity in this new identity. For this to work, it is necessary to enforce proper management of these pseudonyms to maintain private the linkage between identity and its pseudonym [68]. However, current federated models lack an effective mechanism to keep the consistency of users' information during identity modification or revoking [69].

Figure 8 presents a typical federated model, where two organizations establish a federation, meaning that they realize a set of agreements, standards, and technologies to enable both of them to recognize an identity from another organization. Thus, the user identity, which was previously registered by organization 1, is now accepted by organization 2. Hence, the user can now authenticate itself a single time with their IdP and grant access from all SPs that are members of the federation. In

this example, if the user wants to invoke a service from SP in organization 2, the IdP of organization 1 authenticates him and sends a claim-like message to the IdP of the organization 2: *"I am IdP of organization 1, and I authenticate that user"*. Thus, the IdP of organization 1 creates a pseudonym identifier that is linked to the true identity and shares this identifier with IdP from organization 2 to ensure that a user may use SP from organization 2 without disclosing its true identity. Through this pseudonym, both IdPs agree that they are referring to the same user. However, only the IdP from organization 1, which is the one who assigned the pseudonym, stores the user's real identity with all attributes associated with it. The federation model introduces the idea of offering a set of SPs to the user with only a single identity. However, it is unrealistic to assume a global federation to encompass all SPs. Thus, the number of identities that a user needs to manage continues to rise since the user may have an identity on several federations.

There are several federated standards available, which makes IdPs able to exchange identity information. Some of the most popular are Security Assertion Markup Language (SAML), Open Authentication (OAuth), and OpenID.

Security Assertion Markup Language (SAML) [70] was a product developed by the Security Services Technical Committee of OASIS. It is an XML-based framework that enables IdPs to transmit user authentication, entitlement, and identity attributes. In a nutshell, this standard allows two organizations to select and share identity attributes expressed in XML. In a typical use case for a SAML scenario, the user accesses an SP outside her organization and this SP creates and redirects to the IDP where the user is initially registered. This IDP authenticates the user and returns it with a SAML response ensuring user authenticity. The SP verifies the SAML response and, finally, authenticates the user.

OpenID [71] is a decentralized framework for federated (and user-centric, explained in the following subsection) IAM, where a user is capable of accessing different SPs by the Internet through a single digital identity. Therefore, to access some SP, the user first must have an account on any OpenID IdP. After the authentication process, the IdP sends the user a global identifier following a URL format. At this moment, the user can utilize this URL to request services from any SP compatible with OpenID. Under the OpenID framework, SP is entirely dependent on IdP for the user authentication, which means the SP does not have any authentication method to verify the user's identity. Thus, the SP is not able to generate a new unique URL whenever a service is requested to replace the current URL. Unfortunately, since the URL of the user is sent across different SP requests, someone could acquire the authentication URL using a man-in-the-middle attack, getting unauthorized access to an SP. To make things even more complicated, the URL used to identify the user on OpenID is recyclable, meaning that one identifier may become associate with multiple users, leading to the possibility of someone to get unauthorized access to an SP.

OAuth [72], different than OpenID and SAML, is exclusively made for authorization purposes. In a nutshell, with Oauth, the user can delegate the authority to a third-party service (Service hosted at another organization) through tokens, which allows this service to accomplish authorized tasks on behalf of the user. To achieve this, Oauth determines four roles: resource server, resource owner, consumer, and

authorization server. The resource server is a host where the user identity is protected. The resource owner is a host user who owns the data, and authorizes a service to access her identity, respecting the limits imposed by the authorization conceived. The Consumer is a service who wants to access the resource owner identity; however, it must be authorized by the user, and authorization must be validated. The authorization server is an entity that authorizes the consumer to access the resources available on the resource server.

In a typical use case, a service requests authorization to access some information at resource service. If the resource owner authorizes this request, the server receives a token, which determines his authorization limits. Thus, the service asks access to the authorization server by sending its identity and authorization token. If both are valid, the authorization server creates an access token for service, completing the authorization. This token conceives the authorization, and when sent to the resource server, if it is valid, it provides the resource for the service.

User-centric model: This model is the first one to introduce a system that supports identity management at the user side. Instead of managing several identities, a user has a personal tamper-proof device that stores several identifiers and credentials, together with the IdP, who provides them. This device acts like an IdP selector and contains a portfolio of identifiers and credentials from different IdPs. This approach opens up for a user the possibility of only needing to manage its identity with their personal IdP selector. Once authenticated with the IdP selector, the user lets the IdP selector handle the authentication with external IdPs. In this model, on each usage of its identity, a user needs to explicitly approve it, meaning that it is not possible to disclose the information to a third party without her permission. In general, the centralized, federated, and user-centric models put the trust on the IdP, transferring to them control over the identity. Therefore, an IdP becomes a large datastore of personal information, storing all types of data about users.

Figure 9 illustrates a user-centric model. In this example, both IdPs register the user, which consequently has access to both SPs. Instead of memorizing the identifier and credential for each IdP, this task is delegated to the IdP selector, who manages

the authentication process automatically. Once a user authenticates himself to the IdP selector, she can enjoy an SSO experience, without any agreement between these two IdPs.

Self-sovereign model: The Self-sovereign IAMs rely on distributed ledger technology (DLT), which, in essence, is a technological infrastructure and protocols that allow the recording and sharing of data across a distributed network of different participants. It is possible to record, share, and synchronize this data in an immutable manner across the network, without the need of a central coordinator [73]. In short, DLT provides control over the evolution of data between users through a peer-to-peer network, usually using consensus algorithms to ensure the replication among the nodes of the network [74]. The DLT data structure allows the creation of a tamper-proof ledger of transactions, and the ledger remains the same over the network. In short, all participants can view all data recorded on the ledger composed of cryptographically linked “Blocks”, which are digital pieces of information. The security of the DLT came with the fact that once creating and appending a block to the blockchain, it is not possible to change or revert the transactions in that block [75, 76]. Initially focused in the financial sector [77], DLT quickly spread in several fields, inclusively ended up as the key of the self-sovereign IAM [74].

This model emerges from the concept of allowing users to store their identity data, removing any centralized control from the identity authority. Thus, instead of depending on an IdP, users become their own IdP, meaning that they store and manage their attributes. In self-sovereign mode, users are in control of their identity, not relying on a central authority for this purpose. For this to work, identity information must be provided efficiently to those services who need to validate it, must reside in a trusted environment, and it must not be owned or controlled by anyone [78]. For security and privacy reasons, putting any personal data on the ledger is not the best approach since the ledger is immutable. Thus, it is not possible to alter or delete any data written to the ledger. Therefore, instead of sharing current attributes, this model utilizes the DLT to share a set of claims, proofs, and attestations. This model operates through the *Zero-Knowledge Proof* method, which allows one user to prove to another one that they know specific information or meet a specific requirement without exposing the actual information supporting that proof [78].

In the self-sovereign model, three entities are necessary: *Identity Owner*, *Identity Proofer*, and *Identity Verifier*. The *Identity Owner* is a user with complete control over its identity. When the *Identity Owner* desires to share some data with someone else, it turns public the proper information it has. *Identity “Proofer”* is responsible for attesting the validity of the data claimed by the Identity Owner. *Identity Verifier* validates the entity that makes a specific claim and the entity which attested this claim. It is essential to point out that claims made by identity owners can be self-asserted or asserted by another entity whose authenticity can be independently verified by a relying party [78]. We believe that with the popularity of the self-sovereign identity model, the current organizations which actuate as IdPs will have their role redefined as “Identity Proofer”. Instead of storing and managing identities, the organizations will be used only to identify an identity owner and to attest to any claim that an identity owner could make. Through attested data,

both Identity Owner and Identity Verifier experience a safer way to check attributes since that Identity Owner does not share any unnecessary data, and Verifier does not store sensitive data [79].

Figure 10 illustrates the entities of this model. Initially, the user registers itself on DLT, creating a self-generated identification number. This identification number is unique, and no other user knows it. To access some services, the user must make some claims of specific values that they know. In this example, to access SP, the user must have the value of “Attribute 1” more than X . Both SP and the user agree to have the IdP as a trusted party. Thus, instead of sending directly “Attribute 1” to the SP, the user claim that it has “Attribute 1” more than X , and along with the identity number, sends it to Identity Proofer (1). The identity Proofer sends attestation with its digital signature (2), determining “who is the claimer”, “what is the claim”, “who attested it”, “Whether it has been altered” and “Whether the claimer has revoked it”. Now, the user stores and guarantees it is not possible to alter nor delete this attestation on DLT (3). The user requests the service (4) and presents the attestation, leaving Identity Verifier the task to check the digitally signed claim and determine if it came from a relevant authority (5). At this moment, SP can establish a direct, encrypted connection with the user (6), providing the requested service.

4.4 The Seven Laws of Identity and Current IAM Models

Introduced in 2004 by Kim Cameron, *The Seven Laws Of Identity* [80] is a set of principles to which an IAM must conform to offer a universally adopted and sustainable identity system. A metasystem, or system of systems, is a concept of having an IDM that leverage the strengths of all constituent IAM models and provide interoperability among them. This metasystem concept aims to create a consistent IAM system to all of the users, resulting in improvements that benefit all applications, solving several identity systems challenges and making the Internet a safer place. In short, this identity metasystem aims to provide the user identity control when accessing services over the Internet, allowing them to select their digital identity and use them to access the services of their choice. Furthermore, this identity metasystem should enable identities based on different technologies to operate together, where a trusted intermediary that understands both technologies exist and realizes the translations from one technology to another.

In the following, we present each law defined by Kim Cameron [80] and compare the current IAM models according to each law defined, determining which model

approximates to the identity metasystem. Table 1 summarizes the results of this comparison and highlights the laws that each model address.

Table 1 IAMs models in comparison to the laws of IAM.

Law	Isolated	Centralized	Federated	User-centric	Self-Sovereign
Control	-	-	-	-	✓
Minimal disclosure	✓	-	✓	✓	✓
Justifiable Parties	-	-	-	✓	✓
Directed Identity	✓	✓	✓	✓	✓
Pluralism	-	-	✓	✓	✓
Human Integration	✓	✓	✓	✓	✓
Contexts	-	-	-	✓	✓

✓ - Law fulfilled

Law of control: IAM must first offer a convenient and straightforward way to manage user's identities. However, to endure, the IAM should earn the user's trust. Then, IAM must support to put the user in control of the used digital identities and released information. This law is the logic behind **user-centric** and **self-sovereign model**, however not wholly fulfilled by **user-centric**. In **user-centric**, a user still needs a third-party IdP to store identity information, meaning that there is still a need for a third-party IdP, which has some degree of control over her identity.

Law of Minimal Disclosure: There is a risk of a data breach an IAM. The best practice to mitigate this issue is to acquire only the information that a service "need to know", and retain only the information "need to retain". Thus, by following these practices, it is possible to ensure the least possible damage in the event of a breach [80]. In short, except for the **centralized model**, all models have their way of dealing with minimal disclosure. In **isolated model**, all identity's information is contained at one service, meaning that there is no share of this information with other parties [64]. Thus, any data breach is confined at one single service. **Federated model** implements the pseudonymous, which makes data identifiable at third-party IdPs presented at the federation. However, user data is untraceable without accessing the IdP that generate the identity. **User-centric model** has the objective of dealing with this kind of data breach, however, while still having different identities attached at their personal IdP, the identity information still relies on a third-party IdP exposed to the data breach. **Self-sovereign model**, assuming that users should never put the identity data at the ledger, utilizes the zero-knowledge technique, which minimizes the amount of data exposed to data breaches, since it never disclosures the exact identity's data.

Law of Justifiable Parties: IAM must make users aware of the party or parties with whom she is interacting while sharing information. Thus, it represents the central premises of the **user-centric** and **self-sovereign model**. Both **user-centric** and **self-sovereign model** place the users in the middle of the identity process, preserving their freedom to pick their favorite IdPs and to share information.

Law of Directed Identity: To deploy means of managing identities in a hyper-connected world, the IAM must create a relationship between identities, creating a context for a given situation. Thus, the IAM must support two types of identity relationships: "omnidirectional" and "unidirectional". Public entities (IdPs and SPs, for example) should have identifiers that are invariant and well known. These public identifiers can be thought of as beacons –emitting identity to anyone who shows up– relating to anyone in an "omnidirectional" way. On the other hand, when a

user wants to share information with other entities under the law of minimal disclosure, they must create a short-lived relation revealing the least possible identifying information – creating a “unidirectional” identity relationship.

Law of Pluralism: IAM must enable the inter-working of multiple identity technologies run by multiple IdPs. Thus, it means that an IAM must allow the coexistence of multiple technologies. Except for the **centralized** and *isolated model*, all models have tools to operate across different domains. In the **federation model**, when two or more organizations establish a federation, they determine a set of rules and agreements that enables the identity share across multiple models, with each organization implementing their technology. In **user-centric**, the personal IdP offers an interoperability across multiple IdPs. In the Koshutanski *et al.* [81], for example, the authors implement a module on personal IdP that allows users to transform authentication messages from one authentication method to another. In the **self-sovereign model**, the possibility to share identity claims accepted across different services, fulfill the law of pluralism.

Law of Human Integration: IAM must define the human user as a system component. Thus, it must offer security protection for human-device communication to offer protection against identity attacks. Instead of initial identity verification, the user must have other ways to prove its identity. This human integration aspect is attached to the authentication mechanism. It can be achieved, for example, by the integration of multi-factor authentication, meaning that user authentication can only occur when presenting more than one identity verification form. All models can have multi-factor authentication, meaning they all can fulfill human integration at some degree level.

Law of Contexts: In an IAM, a user must be able to interact in an identity relationship, deciding which identity elements to share. Thus, the IAM must incorporate this decision interaction with users. This approval of identity use is the key feature of the **user-centric model** and **self-sovereign model**. By adopting the user-centric model, a user can decide to share an identity from one IdP to another, picking which identity information we share. In **self-sovereign model**, once a user has full control over her identity, she determines which claims to share.

5 IoT on the future of Identity and Access Management Systems

The connected nature of IoT introduces a series of security challenges that come up as a barrier to the wide adoption of IoT. Once IoT encompasses a large number of connected devices, with only a few of them designed with security as consideration [5], data breaches can have a cascade effect, which can lead to devastating consequences. In an autonomous self-driving vehicle, for example, the injection of bogus data can cause a fatal accident. In a diabetes treatment application, a false glucose reading can lead the insulin delivery device to wrongly adjust the amount of insulin, threatening the patient’s life. These examples highlight the need for a mechanism to identify devices, sensors, monitors, and manage their access to sensitive and non-sensitive data before sending or receiving information. In IoT, “things” can have the role of both users and SPs, meaning that they are valuable resources that require management encompassing control and audit [11].

Identity and Access Management Systems (IAM) come with mechanisms for identification, authentication, access control, and auditability to “everything” (devices,

people, networks, applications, and others) for the sake of privacy and data usage protection. Thus, IAM systems are crucial to ensure secure communication and must be present on all the layers of any IoT architecture. However, dealing with the overall scale, scope, and complexity associated with IoT devices brings a series of new challenges that IAM systems must overcome.

5.1 Requirements of the Identity and Access Management System for IoT

IAM for IoT is about dealing with a massive amount of users; thus, each user must have at least one unique identity. However, concerning accessibility and usability, people are used to almost instant results, meaning that the user experience must be taken into account when planning an IAM for IoT. When providing an identity for all “things”, the identity provisioning must be as fast as possible, with correct access rights and the de-provisioning must be just as effective as the provisioning, to avoid malicious user of seizing old identities to initiate an attack. The authentication scheme of the IAM must be able to support multi-tiered authentication where users have relationships and require different authentication methods because IoT devices, as general, are by their nature designed to be simple and to conduct a set of specific tasks. As a consequence, most of them lack security features and computational power, which turns the authentication and cryptography operations challenging. Also, due to their simplicity, most IoT devices do not have a proper interface with the user, meaning that traditional authentication methods, like the password-based, may not be directly applicable to IoT. Moreover, with the scale of IoT, manually authenticating on each device must not be the main form of authentication. For access control, the challenge is not granting the right access levels; instead, the main issue is when and why to grant access. Thus, the challenge lies in setting limits for IoT devices and determine what is appropriate for dynamic large-scale scenarios. Therefore, to address this problem, several works [82, 83, 84, 41] focus on understanding the context of device regular operation and making the access control predict or act based on subtle variations of this behavior.

Overall, those challenges show how IoT is influencing the current IAMs to adapt to offer a more specific identity platform for these devices. In a nutshell, managing persons’ and devices’ identities brings a variety of potential problems due to the nature of IoT. Thus, current existing IAM solutions might not fit the IoT domain. In the following, we present a requirement list to design an IAM for IoT, based on the *seven laws of identity* [80] and IoT characteristics.

Scalability (Req.1): IoT compromises of uncountable things connected, which means that IAM for IoT must operate on a massive scale. However, current legacy IAM platforms are not capable of handling these massive scenarios; in fact, most of them are isolated, inflexible, and unable to scale. When developing an IAM for IoT, the system needs to handle hundreds of millions of identities and access validation actions per second [11].

Mobility (Req.2): A large part of IoT devices compromises of portable devices. In a mobile environment, a user has several devices such as smartphones, smart-watches, or fitness trackers. Thus, when an IAM supports mobility, it must ensure that users are still connected to devices even when moving [85]. Thus, no matter where the devices are, they must be served with all IAM operations.

Easy Device Registration, Revocation and Authorization (Req.3): In real-life IoT scenarios, millions of connected devices require repeatable functions of authentication, registration, provisioning, maintenance, and de-provisioning access. Therefore, the IoT device identity management should be easy, fast, automated, and scalable [86].

Flexible Architecture (Req.4): IoT encompasses a wide variety of devices, from sensors, mobile devices, cars until high computational resources. Consequently, these devices operate across a wide range of programming languages and different platforms. Hence, the IAM must be compatible with all devices providing the flexibility to build applications on any platform or language [86].

Adaptive Authentication (Req.5): As pointed before, the IoT devices are not homogeneous. These devices vary in computational power, connectivity, and power requirements. So, each IoT must support at least one authentication method. Therefore, an IAM for IoT should be flexible to support adaptive authentication for a wide range of devices in different scenarios and with different levels of complexity and security requirements [87, 88, 86].

Continuous & Contextual Security (Req.6): IoT dismantles connectivity barriers and opens doors for the creation of a full connected devices' ecosystem. This feature sparks an interest in IoT due to the new pack of opportunities that emerges. However, hackers and malicious users can also exploit this connected ecosystem. To exemplify new attack techniques, DDoS botnets from IoT devices can become a reality, attacks on pacemaker implants, and also on blood pumps. These two examples by themselves already show that IoT devices can cause great harm. In this direction, context-based security for users and devices is critical in securing IoT. Through contextual information such as geographic location, time, device profile, device behavior, the IAM should generate a risk score that relieves or intensifies the authentication process [88].

In IoT, things are interconnected, changing information and context during all time. Even if a device successfully authenticate to a system, how guarantee that it remains validated over time? Most legacy IAM solutions only protect the initial authentication. For this reason, IoT requires higher security. Applying a contextual identity and adaptive risk at the time of authentication at any point during a communication increases the security for all users once the continuous security approach ensures the authenticity of users at all times and can mitigate risk whenever an anomaly is detected, even for previously authenticated users [89, 90].

Relationship Management (Req.7): IoT introduces an environment with high connectivity, allowing each device to have a series of associated relationships. During an IoT device life cycle, it may change hands numerous times, being necessary to store information such as the device manufacturer, the current owners, the previous owners, significant components, special privacy considerations, and security provisions. This information is not only crucial in identifying the connected devices, but it is also critical for security and to deal with the complexities involving device ownership and access. Therefore, the IAM must be able to create complex and dynamic relationships among the IoT, which includes the respective security ramifications.

Privacy & Consent (Req.8): IoT devices are the primary agents to gather a massive amount of data from users. These data can be very personal (user history,

preferences, health status), meaning that if it becomes publicly available, it can expose the user to many risks, so there is an interest in hiding this data. However, data collected to IoT devices might be necessary for statistics and optimization for that user, making users concerned about how their data is shared and used. Thus, to deliver this personalized data share experience, privacy must be prioritized. In this direction, an IAM must give the users the ability to manage privacy preferences, and consent to data sharing in a way that they can control their data access [91, 92].

5.2 Analysis of Identity and Access Management Systems for IoT

In this subsection, we offer a comprehensive analysis of the IAM proposal for the IoT context. This survey offers a broader perspective of IAMs for IoT showing, for each work, the basic concepts, application context, and technical particularities.

5.2.1 Isolated IAMs

- **Identity management framework for cloud-based internet of things:**

Horror *et al.* [93] propose an IAM framework based on the **isolated IAM** model. The authors assume a scenario where human users and IoT devices are capable to indirectly communicate with each other through a service hosted at the cloud. The authors argue that the cloud, due to the processing and storage capability, is a suitable technology to host an SP and to centralize the IdP functions. The proposed IAM framework has two modules: the *identity manager* and the *service manager*. The first module is responsible for **authentication** operations of the IoT devices, human users, and services. The latter defines the **authorization** functions for IoT devices and human users on the services. In short, the proposed framework follows a publisher-subscriber approach, assuming that IoT devices gather information from different kinds of sensors and publish this information to services on which is subscribed. When a human user accesses a service to get the information collected by IoT devices, she must first authenticate to the cloud, and pass to the **authorization** process, which verifies if her subscription status. The framework considers the **mobility** of both IoT devices and human users as **contextual information** to avoid illegitimate access to the services, meaning that the location and the network on which the IoT device or human user is connected is valuable information for the **authorization**. In short, if the IoT device is connected in a valid location and network, it is allowed to publish the information collected and, similarly, if the human user is connected in a valid location and network, she is allowed to read the information collected by IoT devices. As a result, every location and network also has a unique identification. Although the framework describes the identity manager as the module for the authentication operation, the authors do not define any specific method for it. The **authorization**, on the other hand, is well-defined to follow the **Discretory Access control** (implementing an ACL), with each service having a list of IoT devices and human subscribed users, followed by networks and locations where available. The authors define the identity life-cycle, determining that during the **identity provision** phase, each IoT device and human user of the IAM is created and allocated at one location, resulting in an identity

composed of the tuple: a unique identifier, type of human user or device and location identifier. When a human user or IoT device changes location, the **identity maintenance** updates the location identifier. Finally, when any human user or device leaves the IAM, the authors define that this information must reflect in the identity manager module, **deprovisioning** their identity. In this paper, there is no **auditing** module, which led us to assume that the cloud takes this responsibility. In other words, the cloud can see any activities among human users or devices, which expose the identity information privacy to a third-party. This work assumes a scheme where one universal IAM maintains all IoT devices, human users, and services identities. Even if we assume that the cloud technology provides a solution in terms of computational capacity, building a scalable solution for the communication with the cloud at this universal scale is unrealistic in the IoT context. Besides, the **ACL model** is another drawback of this work, once it supports a large centralized system. The rapid growth of services and IoT devices turn this strategy obsolete because more the need for more complex relationships among those human users, IoT devices, and services.

5.2.2 Centralized IAMs

- **Authentication and Access Control in the Internet of Things:**

Liu *et al.* [94] describe an **authentication** and **authorization** method for an centralized IAM, in a scenario where IoT devices and human users are capable to communicate with each other. An identification key procedure establishes secure communication between the IoT devices and their identity. Basically, through an Elliptic Curve Cryptography-Based **authentication**, both users establish a key for their communication. Those users utilize this key in their communication, ensuring it is **secure** and also considering the **possession factor credential**. The proposed work assumes scenarios with a massive number of devices, and the centralization comes with the assumption that every user is pre-registered on a nearby trustworthy access point, denoted as a registration authority. This authority has the role of the IdP and provides computational and storage capacity, assuming the role of the trusted third-party during the **authentication**. Moreover, this authority is also able to keep a historical record of all accesses for **auditing** purposes. **Authorization** happens through a **hierarchical-RBAC**, meaning that exists an inheritance relationship among roles, resulting in a superior role inherit all permissions of the roles below then. This work does not include **Contextual Information** and **consent** mechanisms. Therefore, several works like [95, 96] act in this requirement, where we point out the work of [95], where offer easy authorization is the main point of their work.

- **Security architecture for mobile e-health applications in medication control:**

Gonçalves *et al.* [97] propose a framework for **authentication** and **authorization** of users and devices for healthcare applications, focusing on a remote medication control system for Ambient Assisted Living. Their framework assumes a **mobile** application where physicians fill out the medical prescription, including the dosage and time to take the medication. A **centralized**

e-health system database stores this information, linking with each patient's RFID. An electronic personal health record, which is a mobile device carried by the patient, can retrieve this information. **Authorization** follows an RBAC model, determining two well-defined roles: patients and physicians, with the former having read-only permission, and the latter having read-write permission. Due to the sensitive nature of the medical environment, the authors argue that in case of emergency, if physicians cannot reach the needed information, it can be dangerous to patients' life. Thus, to solve this problem, they present an **adaptive authentication**, where two types of authentication occur. When the user access read-only operations, such as a prescription consultation, it occurs through a simple login and password method. However, when the physicians perform a read-write operation, such as prescription writing, the authors propose a robust authentication protocol based on public-key certificates, stored in smart cards. Both authentication methods rely on a centralized server application, which acts as an IdP and verifies the validity of the received password or certificate. All the accesses must be **auditable** at a centralized server, maintaining a robust log that identifies the user, time of occurrence, and operation performed. In this work, **contextual security** is out of scope, and the system does not present any form of **privacy and consent** about the patient's data.

- ***Non-Intrusive User Identity Provisioning in the Internet of Things:***

Al-Karkhi *et al.* [98] propose **authentication** and **authorization** methods based on user behavior. The authors argue that, since IoT devices are becoming part of people's lives, the way their devices interact with other devices and services, such as smartphones and connected vehicles, follows the user's behavior. Once that the people's time and attention are limited, this work assumes that traditional authentication methods, such as login and biometric scanning, are not adequate for user identification due to the dynamicity of IoT. If we assume users have **mobility** and several **relationships** with services, the model of user identity confirmation on every interaction is inappropriate for IoT devices. Therefore, to get the user's identity, the proposed work tracks the user's behavior and asserts them into their identities. The authors argue that this solution should help users to avoid and control intrusive interactions, reducing users' disturbance in dynamic environments. For example, the IAM can assert user identity by tracking their access from one service to another while being connected at university. Therefore, this work proposes a **centralized IAM** that creates, at any time, an implicit identity attached to a confidence level for each user. The **authorization** occurs through the MAC model, with the IAM determining, for each service, the minimum required confidence level. Thus, the user authenticates on their devices under any authentication method, and every time that user accesses a service, the IAM verifies the respective context and recent activities, checking if that interaction has occurred recently or not. According to the result of this check, the IAM updates the user confidence level. For example, when a user performs a recognized activity such as accessing a particular service at a specific time, the user confidence increases. The authors assume identifying

and recording only the minimum context information to determine the user's confidence level, thus ensuring **privacy**. However, the user does not have any form of **consent** about the recorded data. The main limitation of this proposal is related to the time required to build the IAM's user knowledge and the lack of flexibility inherent by the MAC authorization model that would encourage users to remain with the same behavior.

- ***A flexible authorization architecture for systems of interoperable medical devices:***

Tasali et al. [96] propose an **authentication** and **authorization** framework for the medical environment. This work assumes a scenario that patients' vital signs are continuously monitored, creating a real-time stream of their health conditions. The monitoring system sends information gathered by these sensors to a centralized application, which can be accessed by clinicians at any time. In short, this application offers real-time data of a set of patients, reducing the need of the clinicians to visit each patient physically. The authors argue that both clinicians and the application must have access restricted to the patient's device, thus ensuring privacy. The authorization system follows both RBAC and ABAC models. Specifically, each application has a role defined by a **RBAC authorization model**, and each clinician has their authorization permissions set by the **ABAC authorization model** since it has more flexibility and can carry **context** information, such as location and relationships with patients. Since application and clinician may have different permissions, the authors propose the flexible authorization model, wherein emergencies, the application can inherence the attributes of the clinician. If the clinician needs to access some device, but the application does not have access, the application can temporarily expand its permissions by inherent permissions of that clinician, allowing them to function correctly in emergencies. Once this work does not implement any form of **consent** for the patient side, they are very vulnerable to social engineering attacks. A malicious user can execute a manipulation to trick patients, setting their permissions into the patient's devices, resulting in a device that gives away sensitive information about the patient without any form of consent.

- ***Identity Management for the Internet of Things: A Software-Defined Networking Approach:***

Sadique et al. [99] propose an IAM architecture based on concepts of Software-defined networks. Authors argue that IoT is composed of different objects, and these objects should be able to travel among different networks, regardless of their locations, network providers, and manufacturers. To achieve this, the authors claim the need for a collective identity among different networks. In their proposal, they designed a location-based IdP and spread identities over several locations of the IoT network. Every device implicitly registers at the closest IdP, and this IdP defines the device's identity context. For every new registration, the local IdP forwards this new identity to a global IdP hosted at the cloud. This global IdP has the origin IdP of every identity, and keep track of context changes. When accessing a service, the device **authentication** occurs at the closest IdP. If this IdP has the identity information,

authentication occurs instantly. If not, the IdP forwards the authentication request to the global IdP. This global IdP, responds to the authentication request and updates identity context on their database and at origin IdP to the new IdP context. The new IdP replicates the device's identity information and responds to the authentication request. The centralization at the global IdP is a significant drawback in this context. Once it inheres concepts of software-defined networks, several works show that control actions, such as rule installation, have surprisingly high latency [100]. Therefore, in this proposal, every time a device changes location, the **authentication** process is vulnerable to a high latency.

5.2.3 Federated IAMs

- A federated architecture approach for Internet of Things security:**
 Leo *et al.* [101] propose a **federated** IAM model in the context of the **smart homes**. The authors argue that since IoT becomes a critical element of the people's lives, the provision of adequate security for the IoT must be across multiple domains. However, due to connected devices heterogeneity, it is necessary to have an entity to mediate the communication among devices. Thus, the authors introduce de Secure Mediation Gateway (SGGW), which is a hub to overcome this heterogeneity and provide secure communication among IoT devices in a domain. In their approach, they partition the IoT devices into two groups: intraSMGW and interSMGW. The first one, denoted as the intraSMGW group, is the internal set of devices belonging to a security domain accessible by a single SMGW. The remaining devices constitute the second group. Each SMGW acts as a centralized IAM for their domain. Among SMGWs, there is a federated network, which enables the secure remote access to devices within a domain supervised by a single SMGW. Thus, SMGW is the boundary between intradomain and interdomain. The proposed architecture shows the importance of a federated IAM to have an internal autonomy or centralized unit to overcome the heterogeneity of various devices. However, the architecture does not specify any **Authentication**, **Authorization** or **Auditing** functionality.
- Consolidate the identity management systems to identify the effective actor based on the actors' relationship for the Internet of Things:**
 Majeed *et al.* [102] present an IAM framework that focuses on the relationship between users and devices. The authors argue that once IoT is an interconnected network of users and devices, the traditional interaction from "owner" and "subscriber" does not fulfill the IoT requirements. IoT is a complex network, with several users interconnected on behalf of devices other than their legal owner. Thus, the authors argue that IoT requires to establish the identity of an actual user – called effective actor – behind any communicated device. However, such identification is a challenge since the IoT environment is not always static, meaning that a user can dynamically establish interactions, changes, or disconnections. The framework proposed by the authors is capable of establishing the effective actor identity of **mobile** objects that might

belong to different IAM in the IoT. When a user engages a relationship with some device, it creates an identity composed of user and device identities and their origin IAM. Then, when a user requests a service from an SP, this relationship identity is presented, containing both users and devices attributes in addition to their origin IdP. The SP sends this information to the respective IdP, which checks the attributes with the users and devices origin IdP. Assuming that this IdP has a trust relationship with the IdPs presented by the user and device, it **authenticates** the user and device relationship, and the **authorization** method occurs through the **ABAC model**, with attributes of both user and device. In this work, the **audit** occurs separately on each IAM, which limits a malicious user to corrupt the log files as a whole. However, in this approach, the authors do not present any **consent** mechanism for users and devices.

- **A Federated Lightweight Authentication Protocol for the Internet of Things:**

Santos *et al.* [103] propose a federated identity authentication protocol for IoT. This work argues that current federated IAMs are mostly ill-suited for IoT devices since most of them are build upon the login/password and have weighty protocols. Addressing this problem, the authors present a federated identity authentication protocol tailored to IoT, based on the assumption that IoT devices are generally resource-constrained. Thus, this work adapts the traditional IDM authentication to achieve a federated lightweight **authentication**. In this direction, they replace weighty cryptosystems by the Elliptic Curve Cryptography, which is more suitable for IoT and reduces the message load in the IoT device. This work presents an authentication solution for IoT. The **authorization** and **audit**, are out of this paper scope.

5.2.4 User-Centric IAMs

- **A user-centric identity management for internet of things:**

Butkus *et al.* [95] propose a user-centric IAM focusing on the mobility of human users and IoT devices. This work assumes a scenario where IoT devices are dynamically interacting based on the identities of their owner, where the relationship between the human user determines which IoT devices one can access. In sum, each human user is the owner of a set of IoT devices, and each **relationship** between humans has a role that set which devices a human user can access from another. For example, when a user visits her friends, their **relationship** determines if her devices are allowed to engage in a service to allow communication and collaboration with devices at the visited place. To address the **mobility** issue, the authors assume that when this human user visits a place, the local IAM returns a list of trusted IdPs to the user. When the user picks her origin IdP, the local IAM redirects the authentication request to this origin IdP to validate this visitor. The authors do not define any specific method for this authentication, leaving to the origin IdP the picking of the most suitable method. Upon successful authentication, the origin IdP sends a token to the visitor. The visitor then forwards the token to the service to get access. Once with this token, the **authorization** relies on the **RBAC**

method. The **auditing** is realized in both local IdP as the visiting IdP and **contextual security** is given by the human relationship.

- **Identity Management in E-Health: A Case Study of Web of Things application using OpenID Connect:**

Domenech et al. [104] propose a user-centric and federated IAM focusing on healthcare applications. The proposed architecture assumes a scenario where a patient wears several medical devices, which continuously gather data about that patient's health condition. The authors assume these devices do not have resources to directly share data on the Internet, meaning that they employ a smart gateway located at the network layer of the 3-layer architecture of IoT, to act as a bridge between a device and the Internet. In the proposed solution, they use the OpenID Connect framework to authenticate users and devices and to establish trust relationships among users and other entities. Thus, when a device or user tries to publish data to the SP, the SP indicates the OpenID Connect Provider for the **authentication** process. Once authenticated, the OpenID Connect Provider issues and forwards a token to the SP. Then, based on **ABAC authorization model**, the SP grants the required access. Since this work uses OpenID, it inherits well-know challenges of this platform. For example, if we consider the latency and throughput of the network and the use of computational resources of the devices, the OpenID contributes to high-load environments, resulting in a performance loss [105]. Furthermore, since the access token mechanism of the OpenID Connect protocol utilizes the same token across different requests, a malicious user can acquire this token to use in a man-in-the-middle attack and get unauthorized access to data. This data leak can compromise other systems and be used to follow up attacks.

- **Cloud-based federated identity for the Internet of Things:**

Freemantle et al. [106] propose a model called OAuthing, that aims to provide a federated IAM and consent management for IoT systems. The authors argue that one of the critical issues in the IoT is the concern that a device usually supports only a specific manufacturer's web system. The manufacturer is the one who manages identity, stores data, provides a user web interface, among others. Once that service can be hacked or may go out of business, the authors determine that this model is not trustworthy. To address this problem, the authors propose a model to reduce the amount of information a manufacturer stores. To achieve this, the authors recommend a separation of devices and users IdP. The users IdP is where the **authentication** method occurs, usually with a login page to users present their credentials. The devices IdP is where the manufacturer stores a secure identity (pseudonym of the device's true identity) of devices. The manufacturers issue a device with a default client identity token at production time. When the user buys the device, they present the secure device identity to the device IdP, which presents a choice of user IdPs to the user. Once the system authorizes the user with their existing user IdP, the device IdP refreshes the stored token on the device representing the logical owner of the device. Thus, the device IdP acts like an identity broker, where when a user wants to access a device, it should authenticate with the user's IdP using some federated identity protocol such

as OAuth, OpenID, and SAML. Once authenticated, the device IdP creates a pseudonym to provide **privacy** for the user. In this proposal, they do not share anonymous identity with anyone, and issue all the **authorization** methods following the **Cap-BAC model**, with random tokens that give permission to perform specific actions but do not identify users. When a service requests access to some user's device information, their pseudonyms are associated with a token to route the request to an instance that is specific to that user. Thus, in this model, the manufacturer only knows the original device identity (e.g., MAC address) and the client identity that has been issued by default. While this model solves the problem by limiting the amount of information that they store, the devices IdP, except for the data and the device's true identity, have access to all information of the IAM. Moreover, this work presents only the ownership relationship, with the device inheriting all user's authorizations from its owner. However, this may not address the IoT relationship requirements of IoT complex scenarios.

5.2.5 Self-Sovereign IAMs

- **Blockchain for IoT Security and Privacy: The Case Study of a Smart Home:**

Dorri *et al.* [23] have proposed a solution where smart homes have local DLT (Distributed Ledger Technology) for controlling and auditing communications of devices, privately and securely. The authors argue that DLT overcomes security and privacy challenges in IoT, since creating and appending a "block" to the blockchain is a permanent operation. However, they argue that adopting DLT in the IoT context is not a straightforward solution. The creation of blocks demands a proof-of-work challenge, which is a way for the DLT to establish a "decentralized consensus". In a nutshell, this proof-of-work demands high computational and energy resources; the transaction confirmation suffers from long latencies due to the consent, and the broadcast of blocks to the whole network is not a scalable solution. Thus, to solve this problem, the authors assume that each smart home is equipped with an always-online, high-end computing device, known as "miner", and has functions similar to the centralized IdP: **authenticating**, **authorizing**, and **auditing** device transactions. However, once that consent solution implies in hard proof-of-work challenges, long latencies, and solutions that do not scale, this work discards these operations, by maintaining a private DLT in a single host. Thus, the "miner" maintains a secure ledger that contains devices activity log, including all requests and their results, without the proof-of-work drawback. In short, this work makes the **accountability** a very hard-to-forge local public log. While this work looks adequate for a single IoT system, the lack of scalability makes it inadequate for large IoT systems. When this work abolishes the proof-of-work challenges using a single server, it completely isolates itself into a silo-like solution, which may not sound a right approach for IoT, once that most of IoT systems are a compromise of a network of systems.

- **Improving the privacy of IoT with decentralized identifiers (DIDs):** Kortensniemi *et al.* [107] propose the use of the self-sovereign identity for IoT devices. They determine that, in less critical applications, the device's identity

can be their IP address or the hardware identifier, such as RFID. However, in critical applications, a device must be able to prove the claimed identity. If this device is used only by its owner, the usage of permanent unique identifiers for the devices present no privacy problems. However, when the owner enables the device to operate with third parties, a permanent unique identifier is a privacy risk, since the system can potentially track information to reveal the device owner. Also, if the device is at some stage sold or borrowed, maintaining the same identifier would put both the old and new owners' privacy at risk. To solve privacy problems, the authors propose the device identity has to be changeable, and one way to implement this is through the self-sovereign identity since it allows them to create, manage, and to discard identities as they seem fitting. The author shows that deploying DIDs directly on IoT lower layers (perception and communication) may not always be possible due to the limited resources and security risks. Thus, the DID must reside in devices that have acceptable computational capabilities, such as gateway or hub devices. In short, this work shows how to introduce DIDs as a complementary function to the OAuth-based **authentication** and **authorization** operations. The **authentication** of devices with the hub occurs through a possession factor, with pre-shared secret keys. When this device communicates with a service, the hub promotes the idea of using anonymous or pseudonymous identifiers for each service and even switching identifiers at suitable intervals, complicating to a malicious user to track and correlate the legit user's activities in different services, thus protecting privacy. In short, this paper shows that DIDs are a suitable solution for privacy-enhancing identifiers of IoT devices. However, not all devices can implement it, being necessary the use of proxy approaches for these more constrained devices.

- **Secure Open Federation of IoT Platforms Through Interledger Technologies - The SOFIE Approach**

Lagutin *et al.* [108] present SOFIE, which is a solution for federating the existing IoT platforms openly and securely using Distributed Ledger Technologies. The authors determine that most IoT platforms and systems are centralized and unable to exchange data among themselves or perform actions across each other. The authors argue that several types of DLTs can be used for IoT platforms, each offering different trade-offs in terms of latency, throughput, and consensus algorithm. Thus, in complex systems, the idea of having a single DLT for everything is unfeasible, highlighting the need to tackle interoperability. To achieve this interoperability, the authors present the inter ledger approach that allows different DLTs to exchange data with each privately. Thus, in their solution, each device is a participant of a private ledger, storing all **authentication**, **authorization**, and **auditing** transactions. When this device wants to share some information with other IoT platforms, it stores only a subset of the data to the main ledger used for collaboration with other ledgers. While this work addresses the **interoperability**, inter ledger operations may take minutes or even hours, which might not be suitable for IoT with real-time restrictions [109].

5.3 Discussion

Table 2 summarizes the works we previously presented, according to the requirements of the Identity and Access Management System in IoT (see Section 5.1). In the following, we further discuss each requirement.

Scalability (Req.1) is highly required at any work that is targeting to present an IAM for IoT because it is one of the most evident requirements for it. Some IAMs models, like the isolated and centralized, have more issues to offer this scalability in relation to the user-centric and self-sovereign. Therefore, instead of only analyzing the model, we will put the technology employed to provide this scalability. Some works, like [93, 106], use the cloud to offer this scalability due to its limitless computing and storage capacity on the paper. However, as pointed out by [110], the centrality of those kinds of solutions may not be the best one when dealing if IAMs. Some works, like [99], avoids this centrality by employing fog computing nodes in their IAM system, to deal with the IAMs in a distributed manner, which can be more appropriate when dealing with IoT. However, some works like [107, 23, 108] take it to another level and employes the DLT as the main technology behind their IAM. While this may offer the maximum scalability and decentralism, the computational power needed to employ this technology may not fit all IoT devices. As a result, we see several works, like [107], that utilize fog nodes as a local centralized unit interacting with the DLT, which creates a local centralized and global distributed solution.

Mobility (Req.2) is another requirement that was present in a significant part of the works analyzed. Since IoT and mobility are several bonded one to another (even if not all IoT devices are mobile by nature) allows some kind of mobility. The work of [93] utilizes the isolated model, which, in theory, is a struggle for mobility. However, once it is employed in Cloud technology, we still consider it mobile. In fact, mobility is so vital that some works, like [95, 99], have this mobility as the main factor when developing their proposal and utilizes it for its own good. It is important to point out that the work [23] utilizes the self-sovereign model, but once it only creates a local DLT unable to interact with the world, we consider it immobile.

Easy Device Registration, Revocation, and Authorization (Req.3) is a gray scenario, once this is not the main topic of several works but yet still present in several of them at some degree. To offer an easy device registration and revocation, we consider that users must be allowed to self provide some essential services, like password recoveries, self-services authorizations, etc. Several works like [95, 96] act in this requirement, where we point out the work of [95], where offer easy authorization is the main point of their work. On the other hand, self-sovereign models shine on this topic because the DLT technology that allows the user to register and revoke their devices in a more natural manner, and the transparency of the DLT allows easy and secure authorization.

In this paper, to evaluate the requirement **Flexible Architecture (Req.4)** we must investigate the IAM model and the technology employed. Isolated models highly tied to the application by nature, which makes them very inflexible. Thus, we consider the work [93] as not flexible enough for IoT. The other models (centralized, federated, user-centric, and self-sovereign), decouple the IAM from the application

and get more flexibility. However, when we observe the technology employed by those IAMs, some particularities can be seen. The ones that utilize cloud computing rely on the concept that resources can be purchased on-demand to satisfy any needs that IAM have, both in terms of scalability and flexibility. However, the work of [36] points out that, even with the unlimited expansion-capacity of the cloud, the distance between the IoT devices and the IAM significantly impacts the latency of authentication and authorization. To offer a proper answer to flexibility problems, the works [111, 112], for example, alleges that cloud has limited flexibility to support IAMs and point out fog computing as the solution of the flexibility and the latency problem originated by the cloud. In [99], the author utilizes fog devices of different platforms to perform the authentication and authorization, which can significantly reduce the delay of those functions and the flexibility of the IAM as a whole. Lastly, when we observe DLT-based IAMs [107, 23, 108], it still in the early stages, with several workings being published recently. However, since that DLT already showed its potential to be used to a wide range of applications and data. We believe that DLT transparency, security, and distributed way to store information and transactions is enough to prove its capacity to offer a flexible and decentralized architecture.

The **Adaptive Authentication (Req.5)** and the **Continuous & Contextual Security (Req.6)** are highly tied one to another. The first one implies that the IAM system must modify the authentication operation to respond to some change in its operating environment. In contrast, the second one implies that the IAM system must consider contextual information to provide better security. Regards to the first requirement, some works take this requirement as a priority and develop the IAM system centered on having some adaptation mechanisms on their authentication operation. The cause of this adaptation, on the other hand, varies a lot from one work to another. Previous works as [98], for example, take the time and location of the user into one university to make the authentication operation adapt based on the safety level required of the place and time. On the other hand, [102] uses the trust relationship of the device's owner to decrease the safety level of the authentication operation to friends when visiting his home. However, as pointed out by the own author of [98], most of the current contextual information is very simple and has some major flaws. The work of [98] concludes that it takes some time to learn the user mobility pattern inside the university to make the authentication mechanism adequate. If this user changes its routine for some reason, the adaptive authentication operation must learn this pattern again. Meanwhile, while the authentication operation does not learn the new pattern, the user will experience several authentications denies, even if he is a legit one. On the other hand, the work of [102] is based on a single feature: the trust of the owner. While this may sound great for the user side, several flaws arise when putting the human factor inside the adaptive authentication. A single human error caused by a misleading over promoting could lead to several data leaks, as we can observe in literature [113]. In short, we see that most of the works that employ adaptive authentication utilize only a few characteristics to adapt or rely on humans to make this adaptation. We believe that exists a lack of a complex solution that utilizes a large amount of context information that IoT nodes could gather.

The **Relationship Management (Req.7)** requirement is the most neglected in our analysis, with only four works addressing it clearly. However, even in those works who implement the relationships, only simple relations are implemented. In [93], the author implemented all IoT relationships and interactions into a simple publisher-subscriber approach. The proposals [98, 106] are the ones who address the IoT relationship more. The first one creates a friend list that changes the authentication based on their friendship and trust, while the second creates some ownership relationships. While both works address those relationships and have this aspect in common, their relationship management is pretty simple and unable to implement more complex relationships as the ones that we present described in Section 2.1. In a Healthcare IoT scenario, for example, if there is a need to implement fine-grained access to patient's IoT devices, this simple publisher-subscriber approach is unable to deliver this need. However, if the whole IAM system already has a net of relationships among devices, users, and applications, enabling fine-grained access is much more straightforward. We believe that this lack of proposals that model complex IoT relationships causes significant problems to enable not only the fine-grained access but also difficult the creation of adaptive authentication mechanisms with the capacity to handle complex situations.

Table 2

Model	Work, Year	Req.1	Req.2	Req.3	Req.4	Req.5	Req.6	Req.7	Req.8
Isolated	Horrow et al. [93], 2012	•	•	•	-	-	-	-	-
	Liu et al. [94], 2012	•	•	-	•	-	-	-	-
	Gonçalves et al. [97], 2013	•	•	•	•	-	-	-	-
Centralized	Al-Karkhi et al. [98], 2014	•	•	•	•	•	Location and Time	-	-
	Tasali et al. [96], 2017	•	•	•	•	•	Emergency	-	-
	Sadique et al. [99], 2020	•	•	-	•	-	-	-	-
	Leo et al. [101], 2014	•	•	-	•	-	-	-	-
Federated	Majeed et al. [102], 2019	•	•	•	•	•	Trust Relationship	•	-
	Santos et al. [103], 2019	•	•	-	•	-	-	-	-
User-Centric	Butkus et al. [95], 2014	•	•	•	•	•	Trust Relationship	•	-
	Domenech et al. [104], 2014	•	•	•	•	-	-	-	-
	Freemantle et al. [106], 2018	•	•	•	•	•	Ownership Relationship	•	•
Self-Sovereign	Dorri et al. [23], 2017	•	•	•	•	-	-	-	•
	Kortesniemi et al. [107], 2019	•	•	•	•	-	-	-	•
	Lagutin et al. [108], 2019	•	•	•	•	-	-	-	•

The **Privacy & Consent (Req.8)** requirement is another one highly tied to the IAM model. We observe that works that utilize the isolated, centralized, and federated model at its pure form, does not offer any mechanism to guarantee the consent of their users. This problem can be partially dealt with in user-centric models by allowing users to choose its IdPs, as observed in some works [95, 104, 106]. However, even if users can choose their IdP, this IdP still represents a third-party that has some control over the users' identities. In theory, the self-sovereign model should offer this extreme consent mechanism by allowing the user to choose the information that he wants to share. Nevertheless, works like [107] that utilize fog nodes (locally centralizes) show that some third party entities can be necessary even in the self-sovereign model.

6 Challenges in Identity and Access Management for IoT

During the past decade, IAMs went from a very narrow and limited research field to an academic and commercial promise to the future of the Internet. With the popularization of IoT, several challenges arise due to the complexity and scale that it brings for IAMs. This scenario boosted research, where several works are aware of these new challenges that IoT introduces. In this section, we present some directions to the future of IAMs.

6.1 Improving the relationship and context identity

In this survey, we show that IoT refers to a hyper-connected world, interconnecting heterogeneous IoT devices in several areas with and without human interaction. This characteristic implies that IoT creates a complex set of relationships, which are not well explored in IAM. The discussions above clearly show that, although much research is going on, this topic still has a lot of open issues. Except for the work of Majeed *et al.* [102], all other works we analyze show the IdP as a flat-file database for their users and devices, with simple relationships such as “owner”, “borrowed” and “network”. We believe that this approach does not address the dynamicity of IoT, and only allows the creation of simple authentication and authorization methods, which may not fit all applications. With more complex relationships and contextual information, the IAMs can better comprehend the device and user behavior, enabling the IAM to use more information to differentiate normal and abnormal behaviour [114]. We suggest that IAMs must go towards graph databases, which is a database that uses graph structures to represent and store data [115]. The use of these techniques should enable the representation of complex relationships and increase IAMs flexibility for IoT. Additionally, this complex data structure, when combined with machine learning, statistical modeling, and predictive analytics, should be able to detect abnormal behavior or even predict when a security breach is about to happen [116].

6.2 Moving towards adaptive security

In this survey, we show that IoT is growing in popularity, and it is connecting various devices, users, and services into a dynamic network. However, the lack of dynamic security mechanisms is still an open issue that impacts the scalability of IAMs [117]. Except for the works of Goncalves *et al.* [97] and Majeed *et al.* [102], the security mechanisms, such as authentication and authorization, are static with limited context visibility. In short, all these mechanisms have a predefined rule which defines the form that a user authenticates and what access she has to a determined service. In IoT, these rules may not be practical to implement since it lacks efficient response strategies in dynamic scenarios [118]. This dynamicity causes several security problems for IoT, such as inappropriate access or invalid authentication denies. We suggest that adaptive security mechanisms should enable the IAM to change its security flow based on the context and relationships of the user (both human and non-human). With this adaptability, the IAM should be able to observe, analyze, and react to IoT dynamically on the fly, adjusting the complexity of security mechanisms to adequate levels based on the users’ relationships and context [118].

6.3 Self-sovereign IAMs in IoT

The self-sovereign IAM model comes with the premise to make users the owners of their identities by providing a unified, interoperable, and tamper-proof data structure. In this survey, we show that a distributed ledger can provide secure connections to support identity operations, such as authentication and authorization. If we take into account the seven laws of identity, we can conclude that the self-sovereign identity model is, in fact, the future of IAM. This distributed ledger enables a massive append-only data structure, where once the system adds data, it is not possible to remove any more. Due to this inability to remove data and the consensus algorithms, the DLT provides trust where there is no trust; once the system publishes a transaction at DLT, it is not possible to refute. The main benefit of the self-sovereign IAM is that no one besides the user is responsible for the protection of their identity data, meaning that the user creates, manages, and utilizes its identities. However, in practice, the DLT demands high computational and energy resources, which makes them unfitting for IoT devices, since most of them are limited in both aspects. Many works rely on centralized devices, which act as central authorities to IoT devices, such as the “Miner” [23] and the private DLT host [108]. Therefore, while the self-sovereign aims to remove any authority for the IAM system, it ends up creating local centralization hosts, which goes in the opposite direction of the main idea of the self-sovereign IAM. Therefore, we believe that, even with we show that DLT is not a silver bullet solution for IAMs in the IoT context, it is not enough to invalidate the benefits this technology enables and further researches must be done in this direction.

7 Conclusion

In this survey, we have presented the concepts, applications, and characteristics of IoT and the concepts and models of IAM systems. To conclude, we present a list of research projects and discuss open research issues. We have explored the literature in an extensive and comprehensible way, discussing the existing works and envisioning the future of IoT and IAM. We observe high demand for the IAM system to be more scalable and more dynamic, driven by the popularization of IoT. We observe that in IoT, access management is more complicated since it needs to understand, not only the right levels of that device but the context of the device and why it is making the request. Among the alternatives proposed recently to complement the current IAM systems, we suggest that if the system should be able to apply machine learning, statistical modeling, and predictive analytics, we can improve access in terms of scalability. We also show that blockchain has the potential to revolutionize the IAMs; however, it is not a silver bullet, since it has serious performance problems, which can end up being isolated self-sovereign IAMs networks that work well for their purpose but does not scale for an IAM that connects all devices of IoT.

In summary, since its creation, the IAMs are in constant change, and the next significant change is to address the requirements that IoT imposes on that systems. The research area still demands more exploration, and we hope this survey inspires future IAM research in the IoT context.

8 Abbreviations

IoT: Mean Internet of Things; IAM: Identity and Access Management; ITU-T: ITU's Telecommunication Standardization Sector; DAC: ACL: Access Control List; Discretionary Access Control; MAC: Mandatory Access Control; RBAC: Role-Based Access Control; ABAC: Attribute-Based Access Control, Cap-BAC: Capability-Based Access Control; SP: Service Provider; IdP: Identity Provider; SSO: Single-Sign on; SAML: Security Assertion Markup Language; OAuth: Open Authentication; XML: Extensible Markup Language; DLT: Distributed ledger technology; DID: Decentralized Identifiers

Competing interests

The authors declare that they have no competing interests.

Author's contributions

Text for this section ...

Acknowledgements

Not applicable.

Author details

¹Universidade Federal do Paraná., ²Universidade Federal de Juiz de Fora., ³Universidade Federal de Viçosa.,

References

- Atzori, L., Iera, A., Morabito, G.: The internet of things: A survey. *Computer networks* **54**(15), 2787–2805 (2010)
- Miorandi, D., Sicari, S., De Pellegrini, F., Chlamtac, I.: Internet of things: Vision, applications and research challenges. *Ad hoc networks* **10**(7), 1497–1516 (2012)
- Statista: Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions) (2019). <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/>
- Abomhara, M., Kjøien, G.M.: Security and privacy in the internet of things: Current status and open issues. In: 2014 International Conference on Privacy and Security in Mobile Systems (PRISMS), pp. 1–8 (2014). IEEE
- Zhang, Z.-K., Cho, M.C.Y., Wang, C.-W., Hsu, C.-W., Chen, C.-K., Shieh, S.: Iot security: ongoing challenges and research opportunities. In: 2014 IEEE 7th International Conference on Service-oriented Computing and Applications, pp. 230–234 (2014). IEEE
- Andaloussi, Y., El Ouadghiri, M., Maurel, Y., Bonnin, J., Chaoui, H.: Access control in iot environments: Feasible scenarios. In: The 9th International Conference on Ambient Systems, Networks and Technologies (ANT 2018), vol. 130, pp. 1031–1036 (2018)
- Torres, J., Nogueira, M., Pujolle, G.: A survey on identity management for the future network. *IEEE Communications Surveys & Tutorials* **15**(2), 787–802 (2013)
- Ranjith, D., Srinivasan, J.: Identity security using authentication and authorization in cloud computing. *International Journal of Computer & Organization Trends* **3**(4), 122–129 (2013)
- Coetzee, L., Eksteen, J.: The internet of things-promise for the future? an introduction. In: 2011 IST-Africa Conference Proceedings, pp. 1–9 (2011). IEEE
- Schulz, P., Matthe, M., Klessig, H., Simsek, M., Fettweis, G., Ansari, J., Ashraf, S.A., Almeroth, B., Voigt, J., Riedel, I., et al.: Latency critical iot applications in 5g: Perspective on the design of radio interface and network architecture. *IEEE Communications Magazine* **55**(2), 70–78 (2017)
- Fongen, A.: Identity management and integrity protection in the internet of things. In: 2012 Third International Conference on Emerging Security Technologies, pp. 111–114 (2012). IEEE
- Harper, R.: *Inside the Smart Home*. Springer, ??? (2006)
- Burton, L., Dave, N., Fernandez, R., Jayachandran, K., Bhansali, S.: Smart gardening iot soil sheets for real-time nutrient analysis. *Journal of The Electrochemical Society* **165**(8), 3157 (2018)
- Banka, S., Madan, I., Saranya, S.: Smart healthcare monitoring using iot. *International Journal of Applied Engineering Research* **13**(15), 11984–11989 (2018)
- Deshkar, S., Thansee, R., Menon, V.G.: A review on iot based m-health systems for diabetes. *International Journal of Computer Science and Telecommunications* **8**(1), 13–18 (2017)
- Uhlemann, E.: Introducing connected vehicles [connected vehicles]. *IEEE Vehicular Technology Magazine* **10**(1), 23–31 (2015)
- Lu, N., Cheng, N., Zhang, N., Shen, X., Mark, J.W.: Connected vehicles: Solutions and challenges. *IEEE internet of things journal* **1**(4), 289–299 (2014)
- Uhlemann, E.: Connected-vehicles applications are emerging [connected vehicles]. *IEEE Vehicular Technology Magazine* **11**(1), 25–96 (2016)
- Patel, K.K., Patel, S.M., et al.: Internet of things-iot: definition, characteristics, architecture, enabling technologies, application & future challenges. *International journal of engineering science and computing* **6**(5) (2016)
- Yaqoob, I., Ahmed, E., Hashem, I.A.T., Ahmed, A.I.A., Gani, A., Imran, M., Guizani, M.: Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges. *IEEE wireless communications* **24**(3), 10–16 (2017)
- Vermesan, O., Friess, P., et al.: *Internet of Things-from Research and Innovation to Market Deployment* vol. 29. River publishers Aalborg, ??? (2014)
- Bertino, E.: Data security and privacy in the iot. In: EDBT, vol. 2016, pp. 1–3 (2016)
- Dorri, A., Kanhere, S.S., Jurdak, R., Gauravaram, P.: Blockchain for iot security and privacy: The case study of a smart home. In: 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), pp. 618–623 (2017). IEEE
- Burhan, M., Rehman, R.A., Khan, B., Kim, B.-S.: Iot elements, layered architectures and security issues: a comprehensive survey. *Sensors* **18**(9), 2796 (2018)

25. Sethi, P., Sarangi, S.R.: Internet of things: architectures, protocols, and applications. *Journal of Electrical and Computer Engineering* **2017** (2017)
26. Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., Zhao, W.: A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet of Things Journal* **4**(5), 1125–1142 (2017)
27. Said, O., Masud, M.: Towards internet of things: Survey and future vision. *International Journal of Computer Networks* **5**(1), 1–17 (2013)
28. Pérez, J.H.: Internet infodiversity: State of the art and future trends. *Qualitative and Quantitative Methods in Libraries* **2**(4), 479–485 (2017)
29. Mendez, D.M., Papapanagiotou, I., Yang, B.: Internet of things: Survey on security and privacy. *arXiv preprint arXiv:1707.01879* (2017)
30. Pal, S., Hitchens, M., Varadharajan, V.: Modeling identity for the internet of things: Survey, classification and trends. In: 2018 12th International Conference on Sensing Technology (ICST), pp. 45–51 (2018). IEEE
31. Clauß, S., Köhntopp, M.: Identity management and its support of multilateral security. *Computer Networks* **37**(2), 205–219 (2001)
32. ITU-T: NGN Identity Management Framework - Recommendation Y.2720. Technical report, ITU-T (2009)
33. Windley, P.J.: Digital Identity: Unmasking Identity Management Architecture (IMA). " O'Reilly Media, Inc.", ??? (2005)
34. Grassi, P., Fenton, J., Lefkovitz, N., Danker, J., Choong, Y.-Y., Greene, K., Theofanos, M.: Digital identity guidelines: Enrollment and identity proofing. Technical report, National Institute of Standards and Technology (2017)
35. Roussos, G., Peterson, D., Patel, U.: Mobile identity management: An enacted view. *International Journal of Electronic Commerce* **8**(1), 81–100 (2003)
36. Shakarami, M., Sandhu, R.: Safety and consistency of subject attributes for attribute-based pre-authorization systems. In: National Cyber Summit, pp. 248–263 (2019). Springer
37. Licehammer, S., Prochazka, M.: Importance of user deprovisioning from services. In: International Symposium on Grids and Clouds (ISGC) 2016, vol. 270, p. 016 (2017). SISSA Medialab
38. Toelen, O.: Identity and access management. PhD thesis, Master. thesis (2008)
39. Bandy, M.T., Mehrj, S.: Directory services for identity and access management in cloud computing. In: 2017 3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), pp. 334–337 (2017). IEEE
40. Tracy, K.: Identity management systems. *IEEE Potentials* **27**(6), 34–37 (2008)
41. Ng, A.C.K.: Contemporary Identity and Access Management Architectures: Emerging Research and Opportunities: Emerging Research and Opportunities. IGI Global, ??? (2018)
42. De Clercq, J.: Single sign-on architectures. In: International Conference on Infrastructure Security, pp. 40–58 (2002). Springer
43. Witkowski, A., Santin, A., Abreu, V., Marynowski, J.: An idm and key-based authentication method for providing single sign-on in iot. In: 2015 IEEE Global Communications Conference (GLOBECOM), pp. 1–6 (2015). IEEE
44. Rashid, A.M., Albert, I., Cosley, D., Lam, S.K., McNee, S.M., Konstan, J.A., Riedl, J.: Getting to know you: learning new user preferences in recommender systems. In: Proceedings of the 7th International Conference on Intelligent User Interfaces, pp. 127–134 (2002)
45. Railkar, P.N., Mahalle, P.N., et al.: Proposed identity and access management in future internet (iamfi): a behavioral modeling approach. *Journal of ICT Standardization* **2**(1), 1–36 (2014)
46. Zhu, X., Badr, Y.: Identity management systems for the internet of things: A survey towards blockchain solutions. *Sensors* **18**(12), 4215 (2018)
47. Jøsang, A., Zomai, M.A., Suriadi, S.: Usability and privacy in identity management architectures. In: Proceedings of the Fifth Australasian Symposium on ACSW frontiers-Volume 68, pp. 143–152 (2007). Australian Computer Society, Inc.
48. Torres, J., Macedo, R., Nogueira, M., Pujolle, G.: Identity management requirements in future internet. ResearchGate (2012)
49. Sandhu, R.S., Samarati, P.: Access control: principle and practice. *IEEE communications magazine* **32**(9), 40–48 (1994)
50. Bosworth, K., Lee, M.G., Jaweed, S., Wright, T.: Entities, identities, identifiers and credentials—what does it all mean? *BT Technology Journal* **23**(4), 25–36 (2005)
51. Trnka, M., Cerny, T., Stickney, N.: Survey of authentication and authorization for the internet of things. *Security and Communication Networks* **2018** (2018)
52. Ometov, A., Petrov, V., Bezzateev, S., Andreev, S., Koucheryav, Y., Gerla, M.: Challenges of multi-factor authentication for securing advanced iot applications. *IEEE Network* **33**(2), 82–88 (2019)
53. Habib, K., Leister, W.: Context-aware authentication for the internet of things. In: Elev. Int. Conf. Auton. Syst. Fined, pp. 134–139 (2015)
54. Jin, X., Krishnan, R., Sandhu, R.: A unified attribute-based access control model covering dac, mac and rbac. In: IFIP Annual Conference on Data and Applications Security and Privacy, pp. 41–55 (2012). Springer
55. Ubale Swapnaja, A., Modani Dattatray, G., Apte Sulabha, S.: Analysis of dac mac rbac access control based models for security. *International Journal of Computer Applications* **104**(5), 6–13 (2014)
56. Sandhu, R.S.: Lattice-based access control models. *Computer* **26**(11), 9–19 (1993)
57. Ferraio, D.F., Sandhu, R., Gavril, S., Kuhn, D.R., Chandramouli, R.: Proposed nist standard for role-based access control. *ACM Transactions on Information and System Security (TISSEC)* **4**(3), 224–274 (2001)
58. Ausanka-Cruess, R.: Methods for access control: advances and limitations. Harvey Mudd College **30**1, 20 (2001)
59. Hu, V.C., Ferraio, D., Kuhn, R., Friedman, A.R., Lang, A.J., Cogdell, M.M., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K., et al.: Guide to attribute based access control (abac) definition and considerations

- (draft). NIST special publication **800(162)** (2013)
60. Gusmeroli, S., Piccione, S., Rotondi, D.: Iot access control issues: a capability based approach. In: 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, pp. 787–792 (2012). IEEE
 61. Lopez, J., Rubio, J.E.: Access control for cyber-physical systems interconnected to the cloud. *Computer Networks* **134**, 46–54 (2018)
 62. Zhang, Y., Wu, X.: Access control in internet of things: A survey. *arXiv preprint arXiv:1610.01065* (2016)
 63. Benantar, M.: Introduction to identity-management models. *Access Control Systems: Security, Identity Management and Trust Models*, 40–72 (2006)
 64. L'Amrani, H., Berroukech, B.E., El Idrissi, Y.E.B., Ajhoun, R.: Identity management systems: Laws of identity for models 7 evaluation. In: 2016 4th IEEE International Colloquium on Information Science and Technology (CiSt), pp. 736–740 (2016). IEEE
 65. Stobert, E., Biddle, R.: The password life cycle: user behaviour in managing passwords. In: 10th Symposium On Usable Privacy and Security ({SOUPS} 2014), pp. 243–255 (2014)
 66. Nida, P., Dhiman, H., Hussain, S.: A survey on identity and access management in cloud computing. *Int. J. Eng. Res. Technol* **3**(4) (2014)
 67. Ma, P., Tao, D., Wu, T.: A pseudonym based anonymous identity authentication mechanism for mobile crowd sensing. In: 2017 3rd International Conference on Big Data Computing and Communications (BIGCOM), pp. 10–14 (2017). IEEE
 68. Carretero, J., Izquierdo-Moreno, G., Vasile-Cabezas, M., Garcia-Blas, J.: Federated identity architecture of the european eid system. *IEEE Access* **6**, 75302–75326 (2018)
 69. Bertino, E., Paci, F., Shang, N.: Digital identity protection-concepts and issues. In: 4th International Conference on Availability, Reliability and Security (2009)
 70. Cantor, S., Moreh, J., Philpott, R., Maler, E.: Metadata for the OASIS security assertion markup language (SAML) V2. 0. OASIS SSTC, March (2005)
 71. Recordon, D., Reed, D.: Openid 2.0: a platform for user-centric identity management. In: Proceedings of the Second ACM Workshop on Digital Identity Management, pp. 11–16 (2006). ACM
 72. Hardt, D., et al.: The oauth 2.0 authorization framework. Technical report, RFC 6749, October (2012)
 73. Rauchs, M., Glidden, A., Gordon, B., Pieters, G.C., Recanatini, M., Rostand, F., Vagneur, K., Zhang, B.Z.: Distributed ledger technology systems: a conceptual framework. Available at SSRN 3230013 (2018)
 74. El Haddouti, S., El Kettani, M.D.E.-C.: Analysis of identity management systems using blockchain technology. In: 2019 International Conference on Advanced Communication Technologies and Networking (CommNet), pp. 1–7 (2019). IEEE
 75. Alharby, M., Van Moorsel, A.: Blockchain-based smart contracts: A systematic mapping study. *arXiv preprint arXiv:1710.06372* (2017)
 76. Lo, S.K., Xu, X., Chiam, Y.K., Lu, Q.: Evaluating suitability of applying blockchain. In: 2017 22nd International Conference on Engineering of Complex Computer Systems (ICECCS), pp. 158–161 (2017). IEEE
 77. Nakamoto, S., et al.: Bitcoin: A peer-to-peer electronic cash system. Working Paper (2008)
 78. Tobin, A., Reed, D.: The inevitable rise of self-sovereign identity. *The Sovrin Foundation* **29** (2016)
 79. Dunphy, P., Petitcolas, F.A.: A first look at identity management schemes on the blockchain. *IEEE Security & Privacy* **16**(4), 20–29 (2018)
 80. Cameron, K.: The laws of identity. *Microsoft Corp* **12**, 8–11 (2005)
 81. Koshutanski, H., Ion, M., Telesca, L.: Towards user-centric identity interoperability for digital ecosystems. *Intl. J. on Advances in Security* **1**, 26–38 (2008)
 82. Anggorojati, B., Mahalle, P.N., Prasad, N.R., Prasad, R.: Capability-based access control delegation model on the federated iot network. In: The 15th International Symposium on Wireless Personal Multimedia Communications, pp. 604–608 (2012). IEEE
 83. Abie, H., Balasingham, I.: Risk-based adaptive security for smart iot in ehealth. In: Proceedings of the 7th International Conference on Body Area Networks, pp. 269–275 (2012)
 84. Atlam, H.F., Alenezi, A., Walters, R.J., Wills, G.B., Daniel, J.: Developing an adaptive risk-based access control model for the internet of things. In: 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCoM) and IEEE Smart Data (SmartData), pp. 655–661 (2017). IEEE
 85. Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., Ayyash, M.: Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE communications surveys & tutorials* **17**(4), 2347–2376 (2015)
 86. Forgerock: The top 12 considerations for an iot ready identity and access management platform. Technical report, Forgerock (2018)
 87. Gebrie, M.T., Abie, H.: Risk-based adaptive authentication for internet of things in smart home ehealth. In: Proceedings of the 11th European Conference on Software Architecture: Companion Proceedings, pp. 102–108 (2017)
 88. Arfaoui, A., Cherkaoui, S., Kribeche, A., Senouci, S.M., Hamdi, M.: Context-aware adaptive authentication and authorization in internet of things. In: ICC 2019-2019 IEEE International Conference on Communications (ICC), pp. 1–6 (2019). IEEE
 89. Bamasag, O.O., Youcef-Toumi, K.: Towards continuous authentication in internet of things based on secret sharing scheme. In: Proceedings of the WESS'15: Workshop on Embedded Systems Security, pp. 1–8 (2015)
 90. Shahzad, M., Singh, M.P.: Continuous authentication and authorization for the internet of things. *IEEE Internet Computing* **21**(2), 86–90 (2017)
 91. O'Connor, Y., Rowan, W., Lynch, L., Heaven, C.: Privacy by design: informed consent and internet of things for smart health. *Procedia computer science* **113**, 653–658 (2017)
 92. Maple, C.: Security and privacy in the internet of things. *Journal of Cyber Policy* **2**(2), 155–184 (2017)
 93. Horrow, S., Sardana, A.: Identity management framework for cloud based internet of things. In: Proceedings

- of the First International Conference on Security of Internet of Things, pp. 200–203 (2012). ACM
94. Liu, J., Xiao, Y., Chen, C.P.: Authentication and access control in the internet of things. In: 2012 32nd International Conference on Distributed Computing Systems Workshops, pp. 588–592 (2012). IEEE
95. Butkus, P., *et al.*: A user centric identity management for internet of things. In: 2014 International Conference on IT Convergence and Security (ICITCS), pp. 1–4 (2014). IEEE
96. Tasali, Q., Chowdhury, C., Vasserman, E.Y.: A flexible authorization architecture for systems of interoperable medical devices. In: Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies, pp. 9–20 (2017)
97. Gonçalves, F., Macedo, J., Nicolau, M.J., Santos, A.: Security architecture for mobile e-health applications in medication control. In: 2013 21st International Conference on Software, Telecommunications and Computer Networks-(SoftCOM 2013), pp. 1–8 (2013). IEEE
98. Al-Karkhi, A., Al-Yasiri, A., Jaseemuddin, M.: Non-intrusive user identity provisioning in the internet of things. In: Proceedings of the 12th ACM International Symposium on Mobility Management and Wireless Access, pp. 83–90 (2014)
99. Sadique, K.M., Rahmani, R., Johannesson, P.: Identity management in internet of things: A software-defined networking approach. In: Proceedings of the 2nd International Conference on Communication, Devices and Computing, pp. 495–504 (2020). Springer
100. He, K., Khalid, J., Das, S., Gember-Jacobson, A., Prakash, C., Akella, A., Li, L.E., Thottan, M.: Latency in software defined networks: Measurements and mitigation techniques. In: Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp. 435–436 (2015)
101. Leo, M., Battisti, F., Carli, M., Neri, A.: A federated architecture approach for internet of things security. In: 2014 Euro Med Telco Conference (EMTC), pp. 1–5 (2014). IEEE
102. Majeed, A., Al-Yasiri, A.: Consolidate the identity management systems to identify the effective actor based on the actor relationship for the internet of things. In: Third International Congress on Information and Communication Technology, pp. 755–765 (2019). Springer
103. Santos, M.L., Carneiro, J.C., Franco, A.M., Teixeira, F.A., Henriques, M.A., Oliveira, L.B.: A federated lightweight authentication protocol for the internet of things. arXiv preprint arXiv:1907.05527 (2019)
104. Domenech, M.C., Comunello, E., Wangham, M.S.: Identity management in e-health: A case study of web of things application using openid connect. In: 2014 IEEE 16th International Conference on e-Health Networking, Applications and Services (Healthcom), pp. 219–224 (2014). IEEE
105. Blazquez, A., Tsiatsis, V., Vandikas, K.: Performance evaluation of openid connect for an iot information marketplace. In: 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), pp. 1–6 (2015). IEEE
106. Fremantle, P., Aziz, B.: Cloud-based federated identity for the internet of things. *Annals of Telecommunications* **73**(7-8), 415–427 (2018)
107. Kortessniemi, Y., Lagutin, D., Elo, T., Fotiou, N.: Improving the privacy of iot with decentralised identifiers (dids). *Journal of Computer Networks and Communications* **2019** (2019)
108. Lagutin, D., Bellesini, F., Bragatto, T., Cavadenti, A., Croce, V., Kortessniemi, Y., Leligou, H.C., Oikonomidis, Y., Polyzos, G.C., Raveduto, G., *et al.*: Secure open federation of iot platforms through interledger technologies-the sofie approach. In: 2019 European Conference on Networks and Communications (EuCNC), pp. 518–522 (2019). IEEE
109. Paavolainen, S., Nikander, P.: Interledger demo: Iot integration. In: 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), pp. 7–8 (2019). IEEE
110. Kim, H., Wasicek, A., Mehne, B., Lee, E.A.: A secure network architecture for the internet of things based on local authorization entities. In: 2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud), pp. 114–122 (2016). IEEE
111. Yi, S., Li, C., Li, Q.: A survey of fog computing: concepts, applications and issues. In: Proceedings of the 2015 Workshop on Mobile Big Data, pp. 37–42 (2015). ACM
112. Wazid, M., Das, A.K., Kumar, N., Vasilakos, A.V.: Design of secure key management and user authentication scheme for fog computing services. *Future Generation Computer Systems* **91**, 475–492 (2019)
113. Pitoglou, S., Giannouli, D., Costarides, V., Androutsou, T., Anastasiou, A.: Cybercrime and private health data: Review, current developments, and future trends. In: Encyclopedia of Criminal Activities and the Deep Web, pp. 763–787. IGI Global, ??? (2020)
114. Yerpude, S., Singhal, T.K.: Internet of things based customer relationship management—a research perspective. *International Journal of Engineering & Technology* **7**(2.7), 444–450 (2018)
115. Robinson, I., Webber, J., Eifrem, E.: Graph Databases. " O'Reilly Media, Inc.", ??? (2013)
116. Meidan, Y., Bohadana, M., Shabtai, A., Guarnizo, J.D., Ochoa, M., Tippenhauer, N.O., Elovici, Y.: Profiliot: a machine learning approach for iot device identification based on network traffic analysis. In: Proceedings of the Symposium on Applied Computing, pp. 506–509 (2017)
117. Mahmoud, R., Yousuf, T., Aloul, F., Zualkernan, I.: Internet of things (iot) security: Current status, challenges and prospective measures. In: 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), pp. 336–341 (2015). IEEE
118. Aman, W.: Adaptive security in the internet of things. PhD thesis (April 2016)

Figures

Figure 1

Relationship among subjects, their identities and attributes.

Figure 2

Identity Lifecycle Framework [33].

Figure 3

Roadmap for the survey, indicating which topics are covered in the literature of Identity and Access Management.

Figure 4

Authentication, Authorization and Auditing.

Figure 5

Identity and access management models

Figure 6

Isolated Model

Figure 7

Centralized Model

Figure 8

Federated Model.

Figure 9

User-Centric Model.

Identity and Access Managment		
Authentication	Authorization	Models
Knowledge	Discretionary	Isolated
Possesion	Mandatory	Centralized
Inherence	Role-based	Federated
Context-Aware	Attribute-Based	User-Centric
		Self-Sovereign

Figure 10

Self-Sovereign model.