Identity Identification and Management in the Internet of Things

Zina Houhamdi[1] and Belkacem Athamena[2]
[1]Software Engineering Department, College of Engineering, Al Ain University, UAE
[2]Business Administration Department, College of Business, Al Ain University, UAE

Abstract: *Henceforth, users agreed on the necessity of continuous Internet connection independently of the place, the manner, and the time. Nowadays, several elite services are accessible by people over the Internet of Things (IoT), which is a heterogeneous network defined by machine-to-machine communication. Despite the fact that the devices are used to establish the communication, the users can be considered as the actual producers of input data and consumers of the output data. Consequently, the users should be viewed as a smart object in IoT; therefore, user identification, authentication, authorization are required. However, the user identification process is too complicated because the users are worried to share their confidential and private data. on the other hand, this private data should be used by some of their devices. Accordingly, an equitable mechanism to identify users and manage their identities is necessary. In addition, the user plays an extreme important role in the establishment of rules needed for identity identification and in ensuring the continuity of receptive services.The main purpose of this paper is to develop a new framework for Identity Management System (IdMS) for IoT. The primary contributions of this paper are: the proposition of a device recognition algorithm for user identification, the proposition of a new format for the identifier, and a theoretical framework for IdMS.*

Keywords: *Authentication, identification algorithm, identity management, internet of things, single thing sign-on, heterogeneous.*

Received February 22, 2020; accepted June 9, 2020
https://doi.org/10.34028/iajit/17/4A/9

1. Introduction

The internet is the only interlinked system that allows global communication between devices through several standard protocols and connections of different types of networks (government, business, academic, etc.,). In the beginning, the internet allowed communication via emails and represented as static websites. But in this day and age, there are multiple implementations of the internet, which is observed anywhere in several forms of life as a set of provided applications and services by meeting users' requirements regardless of the location and the time. This is a direct consequence of the user and mechanisms digitalization [3].

The internet technology demand is reflected in each device of users. It is becoming portable and getting close to the user more than everything and more than ever. Nowadays, smart devices provide a continuous worldwide connection, and this connection becomes compulsory in quotidian lives. Because of the increasing number of connected devices, there is a necessity of a mechanism allowing autonomous communication between devices [11]. The IoT is considered as an encouraging solution. IoT is a network allowing direct interaction between devices using a unique identifier for searching for information. The output of Machine-to-Machine (M2M) communication corresponds/generated to/by users [21]. Moreover, the users are the proprietor of information; therefore, user identification and authentication, secure communication establishment, and resources access are essential.

The users represent a main component of the Internet of Things (IoT), and thus, they are seen as smart things that create, gather and manage information using individual or/and common devices, and consequently, users should be identified in the IoT in a similar way like other things (device, sensor, and actuator). Users are closely involved in IoT since they affect the currently omnipresent internet, and they make electronic devices and create more appropriate user interfaces. Since the user is a crucial component, his identification is mandatory. This represents an attractive area for investigation in order to find solutions for identity management systems that save effort and time [11].

Technically, the IoT is a system of an infinite set of linked things (such as actuators, sensors, devices) that provide several services through the internet. Consequently, IoT represents a new opportunity for business, devices' implementation, leading to services for users. The existence of several layouts and protocols for sensors and devices interaction in IoT and the lack of uniform solution, show the necessity for managing identities to ensure the success of things interaction model [13]. According to numerous proposed models, user identification and

authentication are dependent on network topology, facilities, and regulations [9].

Depending on the application domain, things are identified by a unique identifier or being a member of a specific class (for example, the thing is a car, no matter which car it is). The identification is required for each thing. Thus, Identity management requires things identification for its ends, regardless of the type of technologies used to provide the applications or services to the user [15]. The collection of environmental and sensor data and user-centered reactive services are enabled by accessing shared linked devices.

This paper proposes a new IdMS to address this problem. The proposed model aims to identify users and provides user-centered services by recognizing things in IoT. The IdMS feature is presented as Single Thing Sign-On. The proposed model limits the study area to IoT and M2M. Particularly, it emphasizes on the IdMS, which considers all things in IoT (for example, people, devices, and non-human interface devices such as sensors and actuators). The identification and authentication procedures and it analyses and recommends some appropriate solutions are described.

Also, we discuss the user identification challenges and suggest a new architecture for Single Thing Sign-On Identity Management System (IdMS), which focuses on the end-user, and this IdMS is a user-oriented service system. The suggested architecture allows recognizing the user and his delegated services by identifying only one of his things (device, sensor, etc.,). Furthermore, we propose a new algorithm called Device Recognition (DR) to identify the user. The DR algorithm is theoretically assessed to prove the concept. The results confirm the relevance of the research topic.

Finally, the IdMS is described. Particularly, the IdMS requirements (user and system requirements) are defined. In consequence, a new IdMS framework that manages the identities of the things is proposed.

2. Identity Overview

2.1. Identification

Identity is the window allowing the user to communicate with his/her objects and to exploit services in the present world. In IoT's context, the identity concept is extended to things. Identity is seen as an endpoint to allow access to endpoints easily and independently of concerned things [14]. The identification process allows users to use and modify data, and also permits the customization of services and interactions [18]. Accordingly, identification in IoT associates attributes to represent an identifier. The attribute is a distinctive property associated with a thing, such as sensors with Radio Frequency Identifier label. Identifier differentiates thing from others, and it depends on the application domain [14]. Since the only purpose of the identifier is to recognize things uniquely, it should be strong. The weak identifier shares its value with other things in the system [5]. The identities are generated, controlled, and secured by the IdMS [19].

2.2. Authentication

Authentication is defined as an identity setting up between connected things (users or devices). As a result of things diversity, there is a need for attack resistance and a trivial solution for authentication. In the following sections, the user and device authentication are described separately in detail.

User Authentication: The authentication process validates the identity submitted by the user to verify if it is authentic or not by requesting credentials. Credential refers to authentication tool or Identity checking. It is a certificate or authentication process phase helping the confirmation of the user's identity concerning system ID (such as network address). The credential is essential for authentication and presents a piece of information (password) or distinctive properties (such as NFC and RFID tags or voice/face recognition). Authentication can have one, or several credentials and credential can be [18]:

- *Something acquired*: the user provides a tangible object containing the user's hidden information required by the authentication process. The tangible object can be a USB stick, a smart card, etc. Thus, no need to memorize the hidden information since it is included in the password. However, how to make sure that it provides the right user identification since users can share the objects, the objects can be lost or stolen [6].
- *Something owned*: The user provides his biometric information that are unique physical and behavioural properties such as voice, face digital image, retina, fingerprint, etc. Even though biometric information is unique and supposed to be unchanged, there is a risk related to intentional or unintentional usage of this information (stolen, copied, or falsified) [14].
- *Something known*: The user provides confidential information (such as username/password, patterns, graphical image). These techniques force the user to memorize the confidential information, which is usually complex to avoid its detection by other users [14].

There are other methods for identification, such as analysis of user's behaviour concerning mouse clicks, navigations, or different patterns. Nevertheless, the behaviour-based methods can be imitated, are irresistible to attacks, and their application is restricted in system security [7]. However, biometrics on user behaviour is hard to reproduce since its capture depends on time, and generally, it produces incorrect outputs [8].

Device Authentication: the authentication of devices is a major issue in IoT due to their importance, and they are omnipresent around us. The devise credentials can be:

- *Device key*: The device stores a hidden password. The user should enter the correct password to confirm user identification (referred to previously as "something known"). Usually, the device authentication is performed automatically (does not require human presence at a specific time) [1].
- *Device property*: represents a behavioural credential or tangible contextual feature (such as signal transmission frequency) that is required to find out the identity of the device. Usually, the stated credential is defined based on the context rather than identity [1].

2.3. Authorization and Accounting

While the authentication process aim is to verify user identity, the authorization process aim is to check if a particular user has the right to use a specific resource (data or device) [18]. The authorization process is executed in decision points according to security policy (i.e., there is a comparison between the authenticated thing (requesting access to the resource) permissions and the resource security policy [16, 18].

There are four types of access control methods [16]:

- *Attribute-Based Access Control*: the identity attributes (instead of the identity itself) provides the key for allowing access to a particular resource. Thus, this technique cannot detect a specific identity. To increase the security aspect, all things, including users, activities are documented and saved. The approach is named accounting, and from a security perspective, it is effective since it is executed regardless of the success or the failure of the authentication process, and it is considered as proof in case of a security investigation.
- *Optional and Compulsory Access Control*: the focal point here is the permissions provider. Usually, in compulsory access control, a primary administrator determines the permissions for each system resource. However, in optional access control, the user, representing the resources' proprietor, establishes the access permissions to resources.
- *Role-Based Access Control*: to manage the permissions assignment, a new layer called role layer is added, and the roles are considered as permission subsets. The access permission is associated with the role instead of a particular resource. Consequently, the resource has multiple roles allowing it to function in response to several permission subsets.
- *Access Control Lists*: this method defines a list of permissions assigned to a resource. This method determines which users are authorized to access to resources, as well as what actions are permitted on particular resources. Generally, an access control matrix represents a consistent technique to declare access permissions in a matrix defining things-resources permissions. The main disadvantage of this approach is the complexity of managing a huge number of resources and things.

3. Use Case

This section represents a scenario illustrating the thing identity usage taken from real-world situations [2,11]. An old user installs multiple identities in his own mobile devices (such as laptops, tablets, smartphones, etc.,) to use them for accessing various services. Because of his age, he needs particular care, and he installs in his home eHealth system, which made of wearable devices and motion detectors sustained by a set of sensors to monitor the surrounding context. The eHealth system supervises the user telehealth (the employment of information technology and digital assets to exploit remote healthcare services and to oversee his healthcare) to promote his autonomous living and, when necessary, to inform the appropriate caretakers and relatives about his status to take proper action. Sometimes, this user visits the hospital and stays for a period of time. Thus, he has to bring his devices but not the home sensors. He can use shared equipment in the hospital. Usually, if he wants to use the hospital WIFI, he asks for a password then he types the password in each personal device.

Furthermore, he activates a set of services needed for supporting his assisted living. And vice versa, the complete user information is recorded in a cloud (including the WIFI password), and normally, these identities can be used by the user once his identity is validated. Again, the user validation needs rewriting his username and password. The good news is that this old user is using an IdMS, which is intelligent and provides user identification automatically by detecting his devices. Thus, he can log in spontaneously, and his devices access the WIFI smoothly. Accordingly, the IdMS offers all responsive services independently of location and time". Figure 1 shows the use-case diagram for the above scenario.

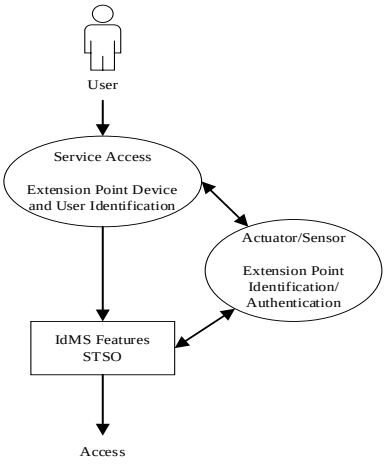

Figure 1. Use-case diagram.

Similarly, the IdMS is useful when this user wants to settle in a hotel or visit his family's home. Thus, all identities are accessible. For instance, he can activate his air conditioner and access his home actuators. Also, he can pay for extra hotel services since the IdMS allows him to access his bank account.

4. Device Recognition Algorithm

This section describes the proposed DR algorithm for identifying the user. It worth mentioning that the proposed *DR* algorithm does not discard the most popular and well-known identification method: the username and password, but it could be viewed as a supplementary method to automate (without user intervention) the authentication process, however, the manual authentication is still optionally used. Figure 2 presents the DR algorithm.

In the beginning, the algorithm assigns for each user a Smart Sheet (SA). SA is a list of identified devices of user A. Note that *SA* is unique. Each device *D* is stored in *SA* list with a unique number within [*1*,..,*m*] (to indicate there are *m* types of devices). Also, for each device type *Dm* there is a set of distinct identifiers types *id*, which is stored in *SA* list, with a unique number within [*1*,..,*n*]. Thus, all device identity types of the user (IdnDm) are recorded in *SA* list.

Whenever an identification request is received from one of the recorded devices in a specific domain, an automatic search starts to identify other user devices and to count the number of available user devices in the local domain. In this case, the user can specify the security level by handling the number of devices required as user identity proof. For instance, the user suggests the identification of 2 out of 4 personaldevices simultaneously to be a rule for his automatic identity identification. Since the user is a part of IoT, he is considered as a rules manager in the system, concerning his preferences.

Then, the algorithm detects and counts the number of available devices of user *A* and recorded in *TA*' sheet. Thereafter, the algorithm calculates the *IA* index representing the ratio between the number of devices in *SA* sheet and the number of devices in *TA* sheet at a given time. Finally, the algorithm checks the needed identification level. In this situation, there are two possible cases: strong and weak identification based on rules of user or services.

- *Strong identification*: The is index (representing Index of Strong identification) claims that all user devices should be available (detected and recognized) except one. Thus, the algorithm compares the *IA* index to *IS* index. If $I \geq IS$, then the user identification succeeds. Otherwise, the algorithm is executed iteratively until the service time-out terminates.
- *Weak identification*: The *IW* index (representing *I*ndex of *W*eak identification) claims that at least half of the user devices should be available (detected and recognized). Thus, the algorithm compares the *IA* index to *IW* index. If $I \geq IW$, then the user identification succeeds. Otherwise, the algorithm is executed iteratively until the service time-out terminates.

Accordingly, the identification rate represented by Equation (1) is used to assess the *DR* algorithm.

$$IA = ND / MD \quad (1)$$

Where, *IA*: defines the coefficient representing the identification rate.
ND: defines the number of identified devices associated with a specific user. Formally, *ND* is the number of devices in *TA*' sheet.
MD: defines the number of entire predefined devices required for user identification. Formally, *MD* is the number of devices in *SA*' sheet.

Since the identification coefficient *IA* of *DR* algorithm rely on the number of recognized user devices, it is mandatory that this coefficient should closer to '1' to identify the user himself.

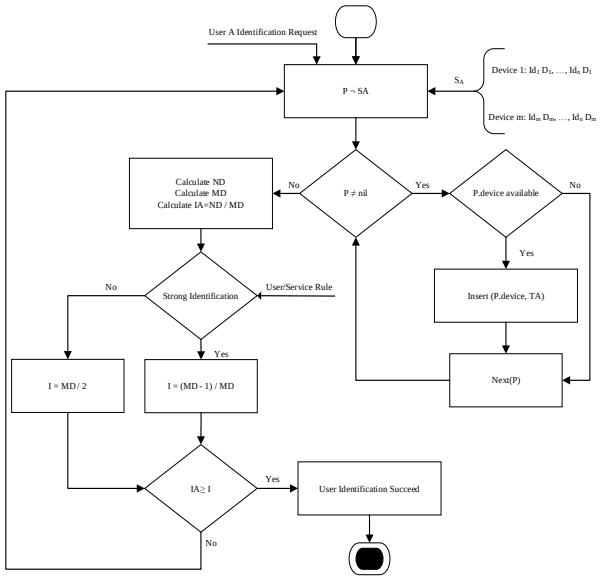

Figure 2. DR identification algorithm.

5. DR Algorithm Analysis

As mentioned previously, the DR-algorithm does not exclude the common and popular login methods. However, the algorithm is new, and it saves effort and time since it presents a computerized method for user identification and authentication. Moreover, the absence of some user devices (lost or stolen) allows the failure of the identification and authentication processes. Consequently, the access to the user's private information or services will be unauthorized, conversely to the password saved on the device. As a precaution against the non-availability of some user's devices at the authentication time, the DR algorithm allows an alternative identification by entering the password (predefined by the user) manually password entry. Thus, the algorithm allows the user's identification in all cases. The suggested identification rate I see Equation (1) allows automatic user identification and authentication.

6. Identification Framework in Heterogeneous IoT Networks

The heterogeneous networks and frameworks are discussed in this section. It starts by summarizing the proposed identification schemes in the literature by listing their advantages and disadvantages. Therefore, we identify research challenges. Subsequently, we introduce a new identifier format to meet the objective of a strong solution for identity management. Finally, we evaluate the suggested identifier format.

6.1. Heterogeneous IoT Networks

IoT is a system of numberless interrelated things (mechanical or digital), actuators, sensors, or merely things following the slogan "everything can be linked to the Internet." The Internet of Things provides an eco-system of services and smart software that are used for improving and simplifying human life and everyday tasks [13]. IoT is a closely related technology to M2M. IoT is set as a foundation to provide and support connections for M2M [17]. The details of M2M architecture is presented in [11].

6.2. Identification Schemes in IoT

In literature, there are several identification schemas Table 1 presents the existing identification schemas in IoT and lists the advantages and disadvantages for each schema [4].

The biggest challenge is related to the integration of various schemes in the IoT structure.

Table 1. Identification techniques comparaison.

Technique	Advantages	Disadvantages
RFID Object Identifier	• Establish code that can adjust any legacy system which is different from GS1 • Addresses several application types by approving the domain code	• Naturally Centralized • No marketing budget for an ISO standard • Does not consider distinct OID structures
Electronic Product Code (EPC) global	• Implements GS1 bar code • Service discovery via End-to-end code • Rapid deployment by major retailers	• Limited to GS1 domain • Restricted and unclear options for RFID data transportation at thing level. • Privacy issue may cause delay and make IoT features superfluous after-sales
Short-OID	• Meet requirements if the complete OID requires to be codified • Needs to codify the OID plus UII (Unique Item Identifier)	• Does not address the OID structure • Similar to RFID OID • Cannot consider the domain-specific differentiation
NFC Forum	• Major investment in infrastructure • Property potential for almost everyone	• Very similar to 2D bar codes • Low data capture integration with other tags • Air protocol-specific
Handle and OID	• A well-established system through a growing number of domains • Number of application can be extended by e-product expenses	• Needs framework overload for supplementary applications • Unsuitable for physical devices • Separated from information transfer
Ubiquitous Code	• Well-developed especially in Japan • TRON used to resolve process, is extremely effective for other systems	• Less strong than EPC global
URL as an identifier	• Propped by browser selection • Aliases are used as "friendly" URL	• Unsuitable for data acquisition • Unsecure
IP address as an identifier	• Allows M2M communication • Appropriate for the majority of IoT devices • Suitable for permanent supervision	• Unsuitable to lightweight M2M • Not Scalable

6.3. Challenges

About today available identification schemes, we identified a list of challenges regarding IdMS for IoT [12]:

- *Creation and management of user and device identities*: the user possesses and uses multiple identities to access various IoT services. Therefore, IdMS must enable the creation of users and devices identities at the time of registration. Then, it introduces them to the authentication process by selecting identities automatically. Hence, the IdMS manages multiple identities relationships to pick the required identities during service access.

- *Devices Authentication*: the first challenge is when a user accesses multiple services simultaneously. The challenge is approached by SAML, OAuth, etc. The access to shared devices by multiple users represents a second challenge which presently addressed by the "sandbox" technique to differentiate users in IdMS for eHealth in IoT. The proposed solution for M2M is defined by authenticating several user's devices, but he is allowed to access only a unique service through all of his devices after executing the authentication of just one device. However, in our proposed solution, the user accesses several services on multiple distinct devices after performing the initial authentication on only one of his devices. Thus, by accessing shared devices, IdMS is able to gather sensors data and contextual metadata and then allows user-centered responsive services.
- *Minimization of human interaction*: In web-applications, Single Sign-On allows access to several services, and consequently, the user interactions for identification and authentication are decreased within heterogeneous networks.
- *Personal devices*: The IdMS should know the proprietor of devices (particularly for users connected to devices) to allow successful communication.
- *Privacy*: Usually, to access one or more services, the user's identities are used. However, certain identities are utilized by only a specific service. Therefore, IdMS must support a technique for managing identity access, which depends on service, which claims access and usage of identity data. Thus, IdMS should not allow sharing user information between different services or at least provide a privacy policy.

6.4. Proposed Identifier Format

A used identifier includes the complete information of the identity for a particular thing. In other words, the identifier defines the domain, user, his device, and non-user interface device uniquely. This identifier assigns the proprietor to things as proposed by Mahalle [14]. Thing in IoT represents users, information, or devices; therefore, it is mandatory to know the thing attributes. Each thing is associated with a unique identifier and set of attributes.

The identifier format proposed by Mahalle is:

$$TI = \langle Thing \rangle \| \langle Thing\ Type \rangle \| \langle Gcontext \rangle \| \langle Lcontext \rangle \| \langle Id \rangle \| \langle CId \rangle$$

Where,*TI*: Thing Identifier.
<*Thing Type*>: indicates the thing type.
<*Gcontext*>: denotes global context.
<*Lcontext*>: denotes local context.
<*Id*>: signifies the thing identifier, which is unique.
<*CId*>: signifies the identity of the context.
The mentioned identifier format is scalable, robust, performant for the one-way delay, and has improved throughput, reduced energy expenditure, and extended lifetime. The identification in IdMS is mandatory. Our suggested thing identifier format, which combines partial identifiers, is:

Devisetype‖*Ginterface*‖*Linterface*‖*DomainId*‖*DeviceId*‖*UserId*

where, *Devisetype*: indicates a partial identifier that defines the type of the device (for example, human user, computer device, actuator, sensor, etc.,).

- *Ginterface*: indicates a partial identifier that defines the global interface or ownership, and it is necessary due to device mobility.
- *Linterface*: indicates a partial identifier that defines the local interface or ownership, and it is necessary due to device location.
- *DomainId*: indicates a partial identifier that defines the domain of thing registration, and it is necessary due to the existence of some domains possessing the same identifier but registered in different Identity Providers.
- *DeviceId*: indicates a partial identifier that defines a unique identifier for each device.
- *UserId*: indicates a partial identifier that defines a unique identifier for each user (device owner) based on a particular domain.

The proposed identifier format aspires to facilitate the user experience by allowing automatic serving that is simpler and effortless than all existing ways for user identification and connection. Although the solution is intuitive and simple, it hides a complex realization and integration. These complexities are hidden and ignore user experience.

The suggested identifier format for things provides telecommunication infrastructure and allows worldwide communication through heterogeneous networks in IoT by considering *Ginterface*. On the other hand, the identifier format contains the partial identifier *Linterface* to allow device portability and localization.

DomainId expresses a single domain to discover the thing easily over several distinct domains. Thus, it allows device portability within heterogeneous networks and also improves the system scalability.

The utilization of *DeviseType* as part of the identifier format ensures the exploitation of different kinds of devices in the system. Moreover, it ensures easy communication if the service needs a particular metric. *DeviceId* distinguishes each device uniquely to provide and utilize information from and to the right device (local or shared). Finally, *UserId* differentiates between users registered in the system. Since each user has his private preferences, services, and devices, this is important for the IdMS to minimize user interactions and simultaneously provide access to the required service by the specified device. In theory, the proposed identifier format is dependable, and the IdMS can use this format to fulfill the requirements

and challenges. It is comparable to Mahalle´s identifier format, and scientifically, both formats are implemented and performed in an identical way.

7. Identity Management System

In this section, the IdMS is described. Moreover, a detailed definition of the system and user requirements, IdMS vision are discussed. Consequently, a theoretical framework for IdMS that addresses the management of thing's identity is proposed. Finally, this section discusses the implementation of the proposed IdMS framework.

7.1. IdMS Description

IdMS is defined as a set of software components and organized hand-operated activities. The purpose of IdMS is the identification and monitoring of computing resources utilization and the support of data integrity and privacy. Furthermore, IdMS involves multiple activities, for instance, generate certificate, manage attribute and role, authenticate and control access, etc. IdMS encloses a set of distributed software components and an enormous number of networking protocols. Moreover, since the IdMS interfaces with business components, its management procedures must conform to business ethics, human resources, and laws regulations. Thus, IdMS design and deployment should consider the cited principles in order to implement successful IdMS. The connections between the IdMS services layer and things layers should be secure, and the access should be controlled. Figure 3 presents the IdMS framework.

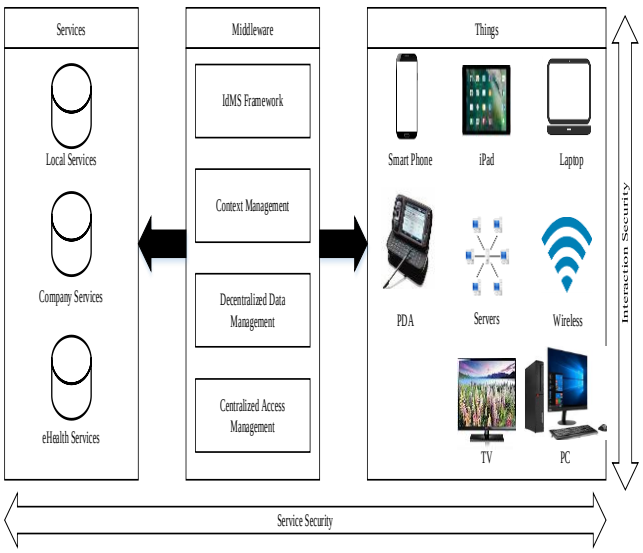

Figure 3. IdMS framework.

In Figure 3, the IdMS architecture contains three layers: services, middleware, and things. Multiple services require collecting and utilizing data from different sources containing data from (internal or external) sensors such as e-health and company. The things are shown as devices with network capabilities which can be high-end (for example, Smartphone and mainframes) or simple devices (such as sensors). The things are distributed and reside in different user-domains, and they cooperate together regardless of their heterogeneity. In the middle of IdMS framework, between the services and things layers, there is a middleware layer, which connects the service and things layers and manages these connections (relationships) insecure manner [14].

7.2. IDMS Models

There are three different types of IdMS models [10]:

- *Centralized IdMS*: In this type of model, the responsibilities and connections are static and rigorously determined. A unique Identity Provider supervises the user authentication, but the information about identities is employed by multiple Service Providers. This model should be reliable for its users and Services Providers. Single Sign-On services are enabled, and consequently, the efforts of users are reduced when they want to benefit from new domain services. The user can create more than one virtual identity. Also, Identity Provider can create multiple independent virtual identities.
- *Isolated IdMS*: In this IdMS model, each service contains its personal identity management. To use this type of model, users create virtual identities, defining part of the complete identity of the user (the entire known information related to the user) required by the user's operations in a specific service domain.
- *Connected IdMS*: The idea of connected IdMS is to manage user identification at the web level and to allow the users to hand out their digital identities throughout different domains. This is referred to as Single Sign-On that minimizes efforts and allows several services access. The development cost is decreased as the Identity Provider performs the authentication**.**

7.3. Identity Management System Requirements

In addition to a set of challenges facing the IdMS, this later is required to address several end-user requirements. The end-user refers to a human user who uses the suggested STSO features. The following summarizes the major IdMS requirements and their analysis based on reviewed literature [13, 15, 18, 20]. Mainly, these requirements influence the incorporation of STSO in IdMS. These requirements are classified in two main categories as follows: User requirements and system requirements.

- *User requirements*: there are two main requirements related to the user that influence the

STSO system incorporated to IdMS. These requirements are identified by considering the end-user needs and expectations from a personnel and technologies perspective:

1. The user owns the rules that operate things: they are part of the IoT eco-system (is set devices connected by a network that communicates with other devices, services, applications, and people). Thus, their role is to inform their requirements, provide feedback, and control the operators separately [18]. The integrated IdMS should personalize the users' profile and accordingly provide a set of services.
2. Continuous receptive services: the system should fulfil and support the users' requirements independently of location and time. The IdMS aims to provide continuous receptive services, depending on particular user' environment and running time, by defining communication mechanisms between things in IoT [15].

- *System requirements*: Five qualities attributes (non-functional requirements) must be considered during the integration of the suggested STSO in IdMS perspective:

1. Security: allows keeping and protecting the private information of the user against illegal and prohibited access.
2. Dependability: allows providing trustworthy interaction on real-time response. Consequently, IdMS must consider the correlation relationships.
3. Extensibility: allows providing virtue and appropriate APIs while integrating new network devices to IdMS.
4. Scalability: allows easy scaling of expected use-cases and discovers new identities (that are unique in different domains) in order to provide worldwide interaction.
5. Flexibility: allows supporting different and several types of system devices.

7.4. IdMS Vision

To meet the expectation of the end-user, the proposed IdMS should be user-centered to achieve the goal of the IoT, which is connecting all things. The IdMS vision is the complete fulfilment of user functional requirements of IdMS for IoT. The IdMS must identify the device by recognizing its unique identifier and automatic sharing of information, characteristics, and competencies between devices. IdMS must allow the interoperability of devices in heterogeneous networks and must minimize user interactions by providing the required and essential mechanisms to authentication.

Figure 4 shows the IdMS vision of the system structure. The structure incorporates multiple computer devices (such as Personal Computers, Smart Televisions, laptops, etc.,) related to different users. These things can access different services by submitting their user's private identities to computer devices. This is necessary to allow the service to recognize the user and to enable the authentication of several services.

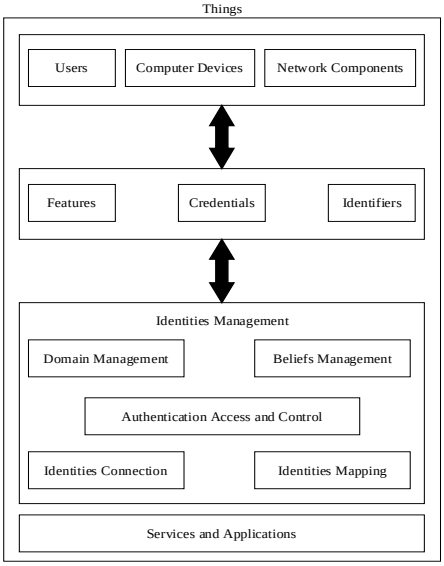

Figure 4. Things access to services.

7.5. Identity Management Architecture in IoT

This paper proposes IdMS architecture, which contains one layer only. The IdMS architecture includes a list of methods for IoT. This IdMS architecture is shown in Figure 5. IdMS manages the identity by authenticating identity and attribute. We introduce a distinct Domain identifier (*DomId*) that depends on the application domain and supports environment knowledge. The main benchmarks in the suggested architecture are related to the domain management, the identities connection and mapping, authentication access and supervision, and finally, lifelong supervision using identities and credentials as inputs [14].

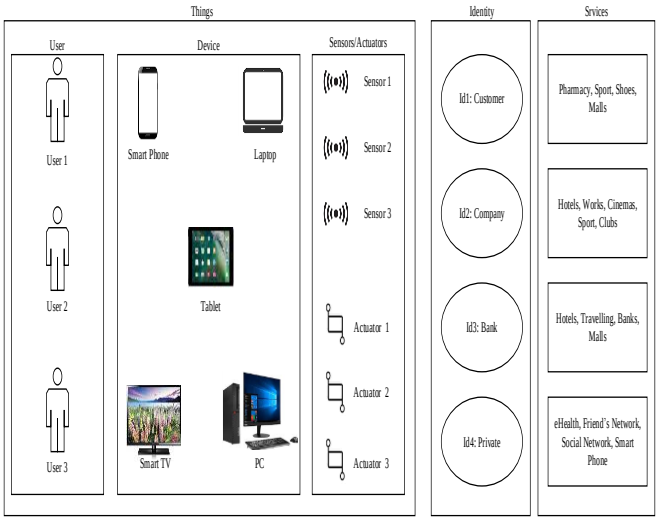

Figure 5. IdMS architecture.

Theoretically, the proposed IdMS architecture seems to be a new encouraging architecture for managing identities in IoT. However, the implementation of IdMS prototype and its validation is necessary for the assessment and analysis of the IdMS efficiency. The attributes of the suggested IdMS depend on Information Communication Technologies that provide data storage, security strategies, specific prototype artefacts and services, heterogeneous network communication, etc. Consequently, a model for information interaction and coordination between different participants is required.

8. Conclusions

This paper overviews the IoT and defines the processes of identification, authentication, and authorization. It describes the importance of intelligent IdMS usage and proposes an appropriate use case scenario to clarify and perceive challenges associated with the identification process. Consequently, a Device Recognition (DR) algorithm is proposed to identify devices automatically and easily. To assess and analyze the DR algorithm, we proposed a factor defining the device identification rate.

In the second part, the paper briefly describes heterogeneous networks and their architecture. Also, we discuss the identification schemes and identifier formats proposed in the literature and identify the research challenges. Finally, we introduce a new identifier format that addresses identity management aspects. A discussion about the evaluation of the proposed identifier format with regards to device type, device mobility, and system scalability is given.

The paper suggested a different and extended view for identity management in IoT to enable computerized communication between different things in IoT by saving user's effort and time.

References

[1] Aboudagga N., Refae M., Eltoweissy M., DaSilva L., and Quisquater J., "Authentication Protocols for Ad Hoc Networks: Taxonomy and Research Issues," *in Proceedings of the 1st ACM International Workshop on Quality of Service and Security in Wireless and Mobile Networks*, Montreal, pp. 96-104, 2005.

[2] Ahmad K., Mohammad O., Atieh M. and Ramadan H., "Enhanced Performance and Faster Response using New IoT LiteTechnique," *The International Arab Journal of Information Technology*, vol. 16, no. 3A, pp. 548-556, 2019.

[3] Alaba F., Othman M., Hashem I., and Alotaibi F., "Internet of Things Security: A Survey," *Journal of Network and Computer Applications*, vol. 88, pp. 10-28, 2017.

[4] Arabsorkhi A., Haghighi M., and Ghorbanloo R., "A Conceptual Trust Model for the Internet of Things Interactions," *in Proceedings of 8th International Symposium on Telecommunications*, Tehran, pp. 89-93, 2016.

[5] Bhargav-Spantzel A., Squicciarini A., and Bertino E., "Establishing and Protecting Digital Identity in Federation Systems," *in Proceedings of the workshop on Digital identity management*, New York, pp. 11-19, 2005.

[6] Corner M. and Noble B., "Protecting Applications with Transient Authentication," *in Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, Mobile System*, New York, pp. 57-70, 2003.

[7] De Luca A., Von Zezschwitz E., Nguyen N. D. H., Maurer M., Rubegni E., Scipioni M., and Langheinrich M., "Back-of-Device Authentication on Smartphones," *in Proceedings of SIGCHI Conference on Human Factors in Computing Systems*, Paris, pp. 2389-2398, 2013.

[8] Feher C., Elovici Y., Moskovitch R., Rokach L. and Schclar A., "User Identity Verification Via Mouse Dynamics," *Information Sciences*, vol. 201, pp. 19-36, 2012.

[9] Finjan Software I., "User Identification and Authentication," 1996.

[10] Haddouti S. and El Kettani D., "Analysis of Identity Management Systems Using Blockchain Technology," *in Proceedings of the International Conference on Advanced Communication Technologies and Networking (CommNet)*, Rabat, pp. 1-7, 2019.

[11] Houhamdi Z. and Athamena B., "User Identification Algorithm based-on Devices Recognition," *in Proceedings of 20th International Arab Conference on Information Technology*, Al Ain, pp. 267-274, 2019.

[12] Kumar V. and Bhardwaj A., "Identity Management Systems: A Comparative Analysis," *International Journal of Strategic Decision Sciences*, vol. 9, no. 1, pp. 63-78, 2018.

[13] Lam K. and Chi C., "Identity in the Internet-of-Things (Iot): New Challenges and Opportunities," *in Proceedings of International Conference on Information and Communications Security*, Singapore, pp. 18-26, 2016.

[14] Mahalle P., Babar S., Prasad N., and Prasad R., "Identity Management Framework Towards Internet of Things (Iot): Roadmap and Key Challenges," *Communications in Computer and Information Science*, vol. 89, no. 2, pp. 430-439, 2010.

[15] Miorandi D., Sicari S., De Pellegrini F. and Chlamtac I., "Internet of Things: Vision, Applications and Research Challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497-1516, 2012.

[16] Rotondi D. and Piccione S., "Managing Access Control for Things: a Capability Based Approach," *in Proceedings of 7th International Conference on Body Area Networks*, Brussels,

pp. 263-268, 2012.
[17] Song J., Kunz A., Schmidt M., and Szczytowski P., "Connecting and Managing M2M Devices in the Future Internet," *Mobile Networks and Applications*, vol. 19, no. 1, pp. 4-17, 2014.
[18] Todorov D., Mechanics of User Identification and Authentication : Fundamentals of Identity Management, Auerbach Publications, 2007.
[19] Trnka M., Cerny T., and Stickney N., "Survey of Authentication and Authorization for the Internet of Things," *Security and Communication Networks*, vol. 2018, no. 7, pp. 1-17, 2018.
[20] Verma P., Verma R., Prakash A., Agrawal A., Naik K., Tripathi R., Alsabaan M., Khalifa T., Abdelkader T., and Abogharaf A., "Machine-to-Machine (M2M) Communications: A Survey," *Journal of Network and Computer Applications*, vol. 66, pp. 83-105, 2016.

Zina Houhamdi received her Ph.D. in Software Engineering. She is an Associate Professor at the Department of Software Engineering, College of Engineering, Al Ain University, UAE. Her research work has been published in several academic journals and has been presented at scientific conferences. Her research areas of interest are data quality, agent-oriented software engineering, software testing, goal-oriented methodology, software modeling and analysis, Petri nets, IoT, and formal methods.

Belkacem Athamena holds a Ph.D. in System Analysis and Design. He is an Associate Professor in the Department of Business Administration, College of Business, Al Ain University, UAE. His main research interest is in system and software modeling and analysis, multi-agent, fuzzy logic, software testing, Petri nets, formal methods, data quality, IoT, and fault diagnosis. He has published many refereed journal articles, contributed chapters and presented papers at conferences.