# Cross-Federation Identities for IoT Devices in Cellular Networks

Bernardo Santos
OsloMet – Oslo Metropolitan University
Oslo, Norway
bersan@oslomet.no

Bruno Dzogovic
OsloMet – Oslo Metropolitan University
Oslo, Norway
bruno.dzogovic@oslomet.no

Boning Feng
OsloMet – Oslo Metropolitan University
Oslo, Norway
boningf@oslomet.no

Van Thuan Do
Wolffia AS
Oslo, Norway
vt.do@wolffia.no

Niels Jacot
Wolffia AS
Helsinki, Finland
n.jacot@wolffia.net

Thanh Van Do
Telenor & OsloMet – Oslo Metropolitan University
Fornebu, Norway
thanh-van.do@telenor.com

*Abstract* — **With the upcoming deployment and usage of Cellular Internet of Things solutions, it is important to mitigate some issues regarding security, due to the physical limitations that some devices have such as low memory, low processing power or even low battery range that don't allow them to perform or support more complex functions. To combine and benefit from the security mechanisms that exist in cellular networks, in this paper we present a cross-federation identity solution that simplifies the authentication and identification procedures for these devices by providing a single sign-on access trait between both the network and the application layers, by also addressing a developed proof-of-concept that allows to illustrate the solution's potential.**

***Keywords — mobile identity management, cross layer identity federation, mobile network security, IoT security, cyber security, cross layer security, cellular networks***

## I. Introduction

As we intend to bring the IoT paradigm to the cellular network - Cellular Internet of Things (**C-IoT**) - there's yet a way for such devices to be secured in a more efficient way that does not cause a burden to those who use them, as we have been noticing an exponential increase on exploits and attacks targeted to those types of devices. Although **C-IoT** devices are equipped with a Subscriber Identity Module (**SIM**) card which could support strong authentication, only weak authentication methods (passwords, pin-codes, etc,…) are still being used by most **IoT** platforms which has been shown that it is insufficient since nowadays we are still experiencing several attacks and exploits that leave them vulnerable (some configurations used are left with their default parameters using widely known credentials, making them easy to breach). It is necessary to provide a different method in a sense that it can be more secure for both devices and platforms while also being more user-friendly and easier to use.

For this purpose, this paper presents a new secure authentication method-based provisioning and management of IoT devices, in which a cross-federation identity between the cellular network and IoT platforms, enabling a unique, stronger and more manageable way to authenticate IoT devices. The paper starts with a brief summary of technologies and procedures that are the basis of this work in II and III. In IV, we describe our proposed solution, with thorough details of the components that make part of it, while in V we provide the implementation details needed to make the solution working and finally, in sections VI and VII are dedicated to evaluations and discussions that were made while developing this work.

## II. Related Work

### A. SIM Authentication

Even nowadays, the cellular network authentication methods based on the Subscriber Identity Module (**SIM**) [1][2] are considered secure and cost effective  and there are attempts to reuse it in the authentication of other applications such as the Generic Bootstrapping Architecture (**GBA**) [3][4] standard, that allows to extend the SIM authentication usage to web applications. It had introduced a new element called Bootstrapping Server Function (**BSF**) that enables authentication of a (mobile) User Equipment (**UE**) by verifying its credentials within the network and, when needed, creates an encrypted session (through the usage of a key created for that purpose) between a UE and a mobile internet application, which is known as Network Application Function (**NAF**) [5].

## III. Identity & Identity Federation

### A. Digital Identity

It is easy to understand and to perceive what we define as an identity by being a list or a set of parameters that characterize who we are. However and nowadays, our identity has multiple meanings to different sectors, and it can be somewhat difficult to unify all that information onto one that satisfies all purposes. The concept of digital identity [6][7][8] tries to overcome this as to provide a way to centralize all means of one's identification into one. Each entity that needs to assert our identification would have access to the needed parameters

978-1-7281-0303-7/19/$31.00 ©2019 IEEE

but wouldn't have the need of isolating them form the core centralized one. This concept can be expanded not only to humans, but also to machines as a way to identify devices in a unique way that would be recognizable in the current trust systems.

### B. Identity Federation

The concept of identity federation can be described as a group or a party that has established common grounds for a procedure, which in this case, on how to be able to identify and recognize a device between the Network and the Application layers. The most known identity federation standards are the following:

#### 1) OAuth 2.0

OAuth 2.0 [8] is a protocol that, by a user's consent, allows a third-party client to access resources (in a server, accessible through the network) on its behalf to facilitate the authentication process. It establishes the concept of an authorization token, providing information on which services on a server the application is authorized to access to, not overriding the access control decision that the server may take.

#### 2) OpenID Connect

OpenID Connect [10] is considered an extension or a profile of OAuth 2.0 rather than a distinct protocol that goes one step further to offer single sign-on and identity provision on the internet. It enables client applications to verify the identity of the user based on the authentication performed by an OpenID Provider, as well as to obtain basic profile information about the user in an interoperable and REST-like manner. OpenID Connect specifies a RESTful HTTP API, using JSON as a data format. Client apps receive the user's identity encoded in a secure JSON Web Token (JWT) called ID token.

## IV. Proposed Solution

As our main objective is to increase the security for **IoT** devices while connected to a cellular network, the following proposal considers essentially adding to the existing network architecture an identity provider and management system [11] as a way to allow **IoT** devices to benefit from the existing secure mechanisms. In this proposal, all devices considered have Universal Subscriber Identifier Module (**USIM**) capabilities. More so, in a way to achieve cross-federation, the same identity that is issued in the network layer will also be recognized by the Application layer, which in this case corresponds to an **IoT** platform. To better understand this, the rest of this section has the purpose of describing the key components of the proposed solution (and its functionalities) as to how they contribute to it.

### A. IoT & Cellular IoT

Internet of Things (**IoT**) is a term that is used to describe the network that allows electronic devices with connectivity capabilities that the ability to exchange data that is usually related to the device's specifications. In fact, supporting all this are known various technologies and protocols such as Machine-to-Machine communications (**M2M**) or Wireless Sensor Networks (**WSN**), among others [12][13].

Cellular Internet of Things (**C-IoT**) expands the prior concept by portraying procedures of having several **IoT** devices performing their data exchanges through mobile network infrastructures without causing usage overload [14][15], and to make that possible, technologies such as Narrowband IoT (**NB-IoT**) and others can be used in order to fulfill this emerging need. Due to the characteristics of these **IoT** devices (mainly cost, size and energy consumption), there is a market for a massive adoption of such components for several types of applications, such as smart homes, e-health, among others [16]. With that, as also referred in [16], it is important to ensure that when such devices need to communicate their readings, there are no disruptions while doing so.

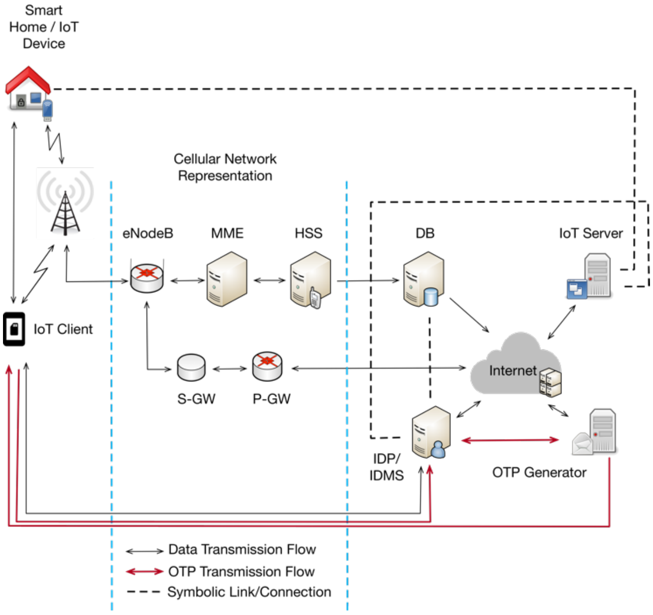

*Fig 1 - Summarized illustration of the proposed solution*

### B. IDP/IDMS Server

Although there's a considerable market for **C-IoT** solutions going alongside with the upcoming deployment of 5G networks, there are some challenges that still haven't been properly tackled, especially when it comes to security. Most of the **IoT** devices deployed in such solutions, due to their specifications, don't have the necessary resources to have proper security mechanisms embedded in order to handle attacks and exploits by themselves, meaning that when it comes to addressing security, a big part relies on the Network and the Application layers to address possible attacks or exploits.

When it comes to the Network layer, there are known mechanisms that allow the devices connected to it to exchange data in a secure way, but when they were developed, they didn't have **IoT** devices in mind, so adaptations must be made. To do so, an Identity and Access Management (**IAM**) system can be integrated so that **IoT** devices can share the same type of identification and authentication strength [1][2][3] as the ones that already benefit from such commodities. By providing such an identity (a set of key-pair values) to an **IoT** device, it allows to strengthen their communications to their corresponding platforms, avoiding or mitigating possible attacks.

The Identity Provider and Management System Server will be the new component added to an existing network infrastructure, which will be in charge of issuing and managing the identities linked to every device that is connected to the network. To do so, it is necessary to reach a consensus on which parameters can be used as identifiers to create such identity that can be perceived by the provider but also by the cellular network and the **IoT** platform.

Firstly, to establish a common ground between the network and the **IDP**, the latter has to know if there's a **SIM** registered to the network that is linked to a certain device, meaning that is has to consult the network's Home Subscriber Server (**HSS**) to obtain that listing. That query however can impose a problem in terms of performance and security for the network, so another component must also be considered, in the form of a secure replica of the **HSS**'s database with only the records that will allow the **IDP** to create the needed identities. This database is isolated from the rest of the infrastructure in the sense that only two secure connections are allowed: First, from the **HSS** itself, a Create, Read, Update and Delete (**CRUD**) interaction is enabled since it will always have the most recent registries and second, a read-only access from the **IDP** as to obtain the needed parameters to be used as identifiers. To facilitate the latter access, an Application Programming Interface (**API**) was created between the secure database and the **IDP** to also help with the creation of the needed identities.

#### *1) Gluu Server*

Gluu Server [17] offers identity management and provision services that allow to establish authentication mechanisms in network resources. More so, it has an integration plugin that allows to sync user information from an Active Directory (**AD**) [18] or Lightweight Directory Access Protocol (**LDAP**) [19] capable servers, which can have access to database information such a list of users or phone numbers. This is called **LDAP** Synchronization [20].

Two other tools are also considered, the System for Cross-Domain Identity Management (**SCIM**) and User Managed Access (**UMA**), which allow to tackle identity management operations in a more simple but efficient and secure way between entities that interact with the issued identities [21][22].

### *C. HSS <-> IDP API Implementation*

To make such identity possible, it is necessary to ensure that such device is registered in the network, meaning that an Identity Provider (**IDP**) has to have an (read-only) access to the network's Home Subscriber Server (**HSS**), so that from its database, the existing records that are linked to the registered devices can be used as identifier parameters for each's device identity. Once created, these identities are managed by an integrated system that allows to log each identities' uses and, in a more dire scenario, blacklist an identity in the case of an exploit or an attack.

In order to facilitate the interaction between the **HSS**' secure replica database and the **IDP** as to achieve federation for the identities issued towards the connected devices, an **API** was developed (*5G4IoT Federation API*). The created API combines the secure read-only access to the database with the **SCIM** tools provided by the Gluu Server, to achieve this first stage of identity federation by performing the following steps:

1. With the mentioned access to the database, it verifies the registries that correspond to connected and active devices in the network and obtains the needed parameters to issue an identity, such as the International Mobile Subscriber Identity (**IMSI**), International Mobile Equipment Identity (**IMEI**), or others such as the Mobile Station International Subscriber Directory Number (**MSISDN**), Packet Data Protocol (**PDP**) type and the **PDP** IP address.

2. After obtaining all sets of registries that fulfil the conditions, the **API** will query the **IDP** as to infer the need of creating or updating an existing identity. The latter case is simpler to consider as it is just an update of the identifiers. However, when no match is obtained, this means that a new identity must be issued with the identifiers that are linked to the corresponding device. With it, and as a fallback access procedure, a randomly generated strong password will be associated to it.

After this assignment, an official issuing request is made to the **IDP**, and if obtained a successful response, this means that an identity has been linked to its corresponding device and can be used for identification and authentication purposes in the network. To summarize the description of the **API** procedure, Fig 2 illustrates it in a flow diagram format.

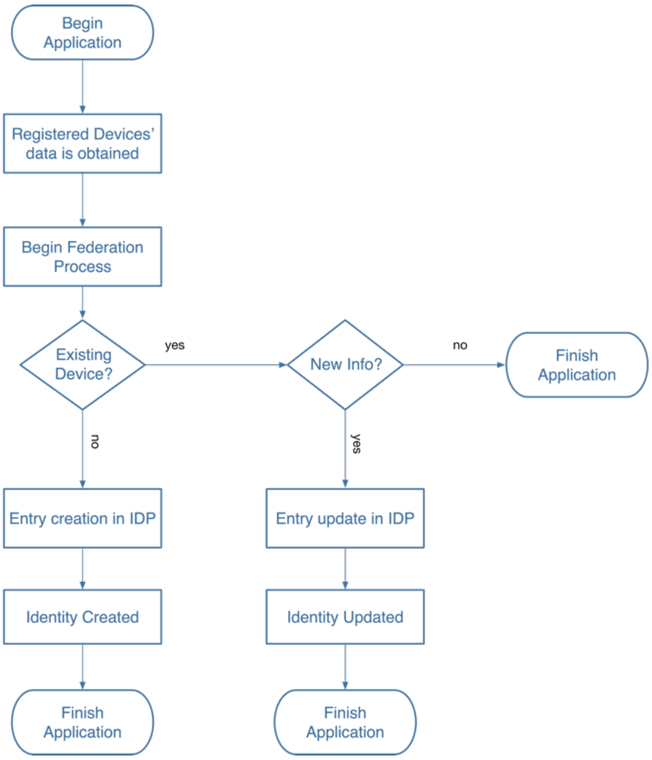

*Fig 2 - API flow diagram*

### D. Network Architecture

The infrastructure is based on OpenStack cloud instituted by Mirantis [23]. On top of it, the Evolved Packet Core (**EPC**) service runs in a Docker [24] container, which connects directly to the Neutron service. From a remote location, we connect the base station – Evolved Node B (**eNB**) through an Edge rack, via public IP. The **eNB** is split into two segments: Remote Radio Unit (**RRU**), i.e. the radio frontend and Baseband Unit (**BBU**), the centralized processing of the radio sampling. The communication between the RRU and the EPC is redirected through the BBU Docker container, residing in the network edge.

The centralized unit utilizes the Calico BGP [25] networking, which can talk to the remote **EPC** deployed in the OpenStack cloud. The mechanism that allows inter-container direct link is the bypassing of layer-2 stacks created by the Open vSwitch [26] (**OvS**) in the OpenStack Neutron. The plugin Kuryr integrates the Docker daemon with the **OvS** by authenticating the virtual machine with the Keystone service. It provides access to the Neutron networking with the specific user that is assigned to. As Kuryr is integrated with the Docker daemon in the VM, now it has relevant control for creating and editing Neutron networks in the specified user.

### E. OTP Generator

Earlier it was mentioned that to each identity that is issued to a device, a randomly strong generated password is created to be used as a fallback access mechanism, however it is not the main access control procedure that the proposed solution considers. In fact, nowadays there still a lot of attacks and exploits in big companies affecting millions of users due to the fact of poor maintenance or even implementation of mechanisms that can reinforce security during access control situations [27][28].

One of those mechanisms is Two-Factor Authentication (**2FA**), in which when a device tries to log into a service, a one-time password (**OTP**) is sent to the device linked to an account to prove its authenticity. To that, another component that is also considered in our proposal is an **OTP** generator as to implement this mechanism.

So, when a device tries to access to a resource in the network, the **IDP** will verify the identity of that device. After that, in a periodic fashion, the **IDP** will perform a request to the **OTP** generator to send a code to the "claimed" device to prove its ownership and authenticity. If the device is whom it claims to be, it will have received the code and has to provide it to the **IDP**. Only then the device will be considered properly verified and deemed trustable to use the network and its resources.

### F. IoT Platform

In order for **IoT** solutions fulfill their purpose, it is detrimental that **IoT** devices are able to communicate their readings to their corresponding platforms/servers without any risk of compromise nor exploit due to attacks that occur in both layers. So far, we have been describing how to mitigate possible situations that can happen in the Network Layer, but in order to provide the same mechanism towards the Application Layer, it is necessary also to apply changes in the **IoT** platforms, as our second stage of identity federation needs to be accomplished between the platform and the IDP.

By now, it has been clarified that devices registered in a cellular network have an identity that will allow them to benefit from the existing secure mechanisms, but this identity has also to be recognized by the platforms. This is achieved by integrating a programmed module from the Gluu Server that will allow a service provider to "understand" these identities. To be more precise, this module will allow any **IoT** platform to, when facing an authentication challenge by one of these devices that have been given an identity, to ask the **IDP** to verify them and tell the platform if they are indeed who they claim to be, meaning that we provide to the platform a single sign-on mechanism, making it easy for devices to log-in and connect to their corresponding platforms, while considering the security aspects to perform that action.

If validated by the **IDP**, the platform will then allow the device to communicate its readings, but if not (due to the case of an exploit), the device no longer can communicate with the platform until the issue has been solved. More so, the platform's device database will also be updated in order to comply with this integration, as it will need to share at least some of the identifiers (from an application only perspective), in order to prompt both the device and the **IDP** with the correct identity.

By deploying these changes, the second stage of federation is completed and so, the creation of cross-federation identities is made possible.

### *G. IoT Client*

A final aspect to account in our proposal is the development of **IoT** clients, as to manage the devices through another type of hardware/software remotely (e.g. smartphones), or simply a middleware linked to a device that manages the connections to its corresponding platform.

Similarly, in F., these clients will also have to have the ability to deal with authentication challenges when interactions between the devices and the platforms occur, and so they must be certified and recognized by the **IDP**, meaning that another module from Gluu Server must be added to the client.

This module will provide a unique identity to the client and the Uniform Resource Locator (**URL**) of the **IDP**, which is also known as the discovery protocol and with it, it allows the **IoT** client to process the challenges mentioned before. This also means that the module will also manage (as a local copy) any authorization tokens that was given to a certain device.

## V. Implementation

To validate the proposed solution, a proof-of-concept is built at the Secure 5G4IoT Lab at Oslo Metropolitan University consisting of a 4G/5G mobile network extended with 3 Identity Management entities and 2 IoT entities (Fig 3).

### *A. Cellular Network*

To create an early implementation of a 5G mobile network *OpenAirInterface* [29], an open source communication software elaborated by EURECOM is first installed in generic computers and then later virtualized on the OsloMet Cloud Infrastructure to achieve a software-based 5G mobile network.

The 4G LTE base station **eNB** is composed by a generic PC running Kali Linux and OpenAirInterface **eNB** connected to an USRP (Universal Software Radio Peripheral) N200, which is a software-defined radio designed and commercialized by Ettus Research [30], and the Evolved Packet Core (**EPC**) including the **HSS** is composed by a generic PC running Ubuntu and OpenAirInterface.

### *B. IoT Entities*

#### *1) IoT Platform/Server*

The IoT Server is composed by a generic PC running Ubuntu, a Gluu 3.1.5 module and also a lightweight M2M server open source using Eclipse Leshan [31].

#### *2) IoT Devices/Clients*

For IoT devices, we use a *Nuki* [32] Lock, that allows to perform more complex interactions due to its developer **API**, and as IoT clients, we use smartphone devices with Android OS equipped with the *AppAuth* [33] Software Development Kit (SDK) in order to interact with the *Nuki* Lock, the **IDP** and the network.

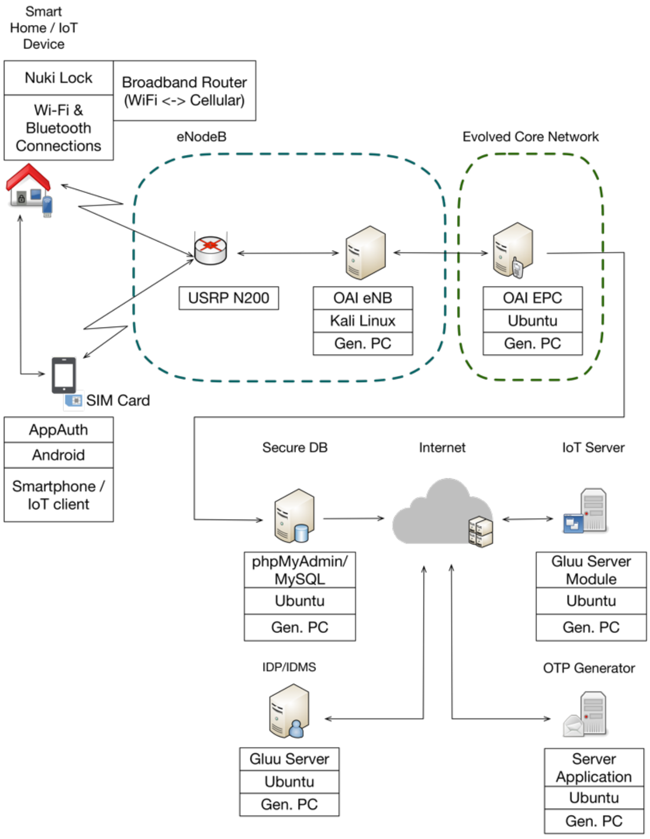

*Fig 3 - Illustration of the Implemented Proof of Concept*

### *C. Identity Management Entities*

#### *1) Identity Provider*

A generic PC running Ubuntu and Gluu Server 3.1.5 [17] is used, which is an open source identity provider server software compliant with OpenID Connect. Our choice relied on the fact of its fast deployment and flexibility and also because it can act not only as an IDP but also as an IDMS which permits us to review and control all the identities to be issued and to handle basic profiling options, such as user grouping and role assignment and also allowing integration options as mentioned in IV.B.1).

#### *2) Secure Replica Database with Federation API*

For the database, we also use a generic PC running Ubuntu and MySQL, an open source relational database management system (**RDBMS**). The Federation API mentioned in IV.C was created using the Java [34] programming language, due to its ease of implementation and interoperability with other components.

#### 3) OTP Generator

For the **OTP** generator, a generic PC running Ubuntu is also used with an application which performs the interactions described in IV.E.

## VI. Evaluation

The proof-of-concept has been tested with the focus on flexibility and usability for IoT applications. The ability of registering and removing new IoT owners and their devices at the ***IDP*** has been verified, as well the creation of a simple **IoT** client with a Gluu module took place, being able to redirect authentication requests to our **IDP**.

We've also tested the performance of our federation **API**, with nearly 50 entries accumulated thus far in our **HSS**, we are able to achieve the first stage of federation process for those entries in 3 to 4 seconds (multiple tries were made, and it always come to those values). It is worth mentioning that when considering a larger number of registries, this process will take a bit longer, but considering that this process will not interfere with the network's performance and availability and only has to occur periodically, it is not considered a downside, but it is a known issue.

## VII. Discussion and Future Work

By presenting the showcased identity federation, we have demonstrated a secure way for devices to use the cellular network and be properly authenticated to their corresponding platforms, however there are some limitations and considerations to take as we progress with our work:

### A. Full integration between IoT Platforms and IDP

Although the Gluu module integration was addressed, further testing to its implementation must be considered as a way to provide a generic solution for all types of platforms, since for this work we've considered and used open-source.

### B. Inclusion of other IoT verticals

As we progress with our work, it is important to consider other types of devices to study their behaviour and performance to extend this solution with multiple **IoT** verticals, for example, when having smart-home and e-health devices integrated in one solution, our proposal should be able to provide identities and manage to all devices, despite their different purposes.

### C. Expand the identity scenarios

In this work, we've discussed that the identities issued are directly correlated to the devices enrolled in the network, however, it is pertinent to consider other types such as for IoT solutions' owners and users and with that also addressing role assignment and group profiling mechanisms. In short, an identity can be used in multiple scenarios and/or with different contexts, so it can be necessary to profile an identity to establish their roles/privileges for each given context.

### D. OTP Alternatives

Although the inclusion of the **OTP** generator gives another layer of access-control security or introduces a **2FA** sequence, this could be addressed with yet another security mechanism existing in cellular networks, more specifically, the Temporary Mobile Subscriber Identity (**TMSI**). This given identity is temporarily assigned to a device when it is going through a location registration and can be reassigned at certain intervals by a mobile operator, meaning that it wouldn't cause any exploits by reusability. As of now, it is not possible to obtain such identity without causing any type of tempering with the **SIM**, hence the OTP generator option is used for now.

## VIII. Conclusion

By achieving the proposed federation, it is possible to have an issued identity that it can be confirmed by the network' services and granting it access to the proper and necessary resources, in this case the data that is obtained from an **IoT** device to the **IoT** platform is considered trustworthy and secure. This will allow to have **IoT** devices connected in such way that, while being completely manageable through their identities, they can be used in various contexts (also known as verticals) that will allow their massive yet secure integration in this upcoming network generation.

Although the feasibility of the solution has been verified, the performed tests are still limited, and more diversified tests are needed in order to cover all the relevant scenarios.

## Acknowledgment

This paper is a result of the SCOTT project (www.scott-project.eu) which has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737422. This Joint undertaking receives support from the European Union's Horizon 2020 research and innovation program and several countries such as Austria, Spain, Finland, Ireland, Sweden, Germany, Poland, Portugal, Netherlands, Belgium and Norway

## References

[1] 3GPP: TS 11.11 Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) Interface, ver 8.14.0, 12-06-2007

[2] 3GPP: TS 31.102 Characteristics of the Universal Subscriber Identity Module (USIM) application ver 16-06-2017

[3] 3rd Generation Partnership Project: 3GPP TS 33.220 V8.2.0 (2007-12) Technical Specification Group Services and System Aspects; Generic Authentication Architecture (GAA) Generic bootstrapping architecture (Release 8)

[4] Timo Olkkonen: Generic Authentication Architecture, Helsinki University of Technology - http://www.tml.tkk.fi/Publications/C/22/papers/Olkkonen_final.pdf

[5] D. Van Thanh, I. Jørstad, and D. Van Thuan, "Strong authentication for web services with mobile universal identity," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9228, pp. 27–36, 2015.

[6] R. Rivera, J. G. Robledo, V. M. Larios, and J. M. Avalos, "How digital identity on blockchain can contribute in a smart city environment," *2017 Int. Smart Cities Conf. ISC2 2017*, vol. 00, no. c, 2017.

[7] M. Lenco, "Digital identity as a key enabler for e-government services," *Mob. Connect - GSMA*, pp. 1–8, 2016.

[8] Maliki, T. El, & Seigneur, J. (2007). A Survey of User-centric Identity Management Technologies Requirements. *International Conference on Emerging Security Information Systems and Technologies*, 12–17. http://doi.org/10.1109/SECURWARE.2007.6

[9] IETF Request for Comments: 6749: The OAuth 2.0 Authorization Framework, October 2012

[10] OpenIDConnect:http://openid.net/connect/

[11] B. Santos, V. T. Do, B. Feng, and T. van Do, "Towards a Standardized Identity Federation for Internet of Things in 5G Networks," in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2018, pp. 2082–2088.

[12] Frustaci, M., Pace, P., & Aloi, G. (2017). Securing the IoT world: Issues and perspectives. In *2017 IEEE Conference on Standards for Communications and Networking, CSCN 2017.* http://doi.org/10.1109/CSCN.2017.8088629

[13] Masek, P., Fujdiak, R., Zeman, K., Hosek, J., & Muthanna, A. (2016). Remote networking technology for IoT: Cloud-based access for AllJoyn-enabled devices. In *Conference of Open Innovation Association, FRUCT.* http://doi.org/10.1109/FRUCT-ISPIT.2016.7561528

[14] J. S. Kim, S. Lee, and M. Y. Chung, "Time-division random-access scheme based on coverage level for cellular internet-of-things in 3GPP networks," *Pervasive Mob. Comput.*, vol. 44, pp. 45–57, 2018.

[15] Z. Ning, X. Wang, X. Kong, and W. Hou, "A social-aware group formation framework for information diffusion in narrowband internet of things," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1527–1538, 2018.

[16] S. Dama, V. Sathya, K. Kuchi, and T. V. Pasca, "A Feasible Cellular Internet of Things: Enabling Edge Computing and the IoT in Dense Futuristic Cellular Networks," *IEEE Consum. Electron. Mag.*, vol. 6, no. 1, pp. 66–72, 2017.

[17] GluuServer–https://www.gluu.org

[18] Active Discovery - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-whatis

[19] LDAP-https://ldap.com/

[20] Gluu LDAP Synchronization – https://www.gluu.org/docs/ce/user-management/ldap-sync

[21] Gluu SCIM 2.0 User Management – https://www.gluu.org/docs/ce/user-management/scim2

[22] Gluu UMA 2.0 Authorization Server – https://www.gluu.org/docs/ce/admin-guide/uma

[23] OpenStack Mirantis - https://www.mirantis.com/

[24] Docker – https://www.docker.com

[25] CalicoBGP-https://www.projectcalico.org/why-bgp/

[26] OpenvSwitch-https://www.openvswitch.org/

[27] S. Boddy and J. Shattuck, "The Hunt for IOT: The Growth and Evolution of Thingbots Ensures Chaos,", F5 Labs, Seattle, Washington, USA, 2018 - https://www.f5.com/labs/articles/threat- intelligence/the-hunt-for-iot-the-growth-and-evolution-of-thingbots- ensures-chaos.

[28] Glenn Fleishman, Equifax Data Breach, One Year Later: Obvious Errors and No Real Changes, New Report Says, Fortune, 07th September 2018: http://fortune.com/2018/09/07/equifax-data-breach-one-year-anniversary/

[29] The OpenAirInterfaceTM Software Alliance (OSA) http://www.openairinterface.org/

[30] Ettus Research, Inc., USRP N200 (Online - Available at: https://www.ettus.com/product/details/UN200-KIT - [Accessed November 2017])

[31] Leshan: https://eclipse.org/leshan/

[32] Nuki Smart Lock: https://nuki.io/en/

[33] AppAuth:https://appauth.io/

[34] Java Language: https://www.java.com/en/