On the design of a decentralized and multi-authority access control scheme in federated and cloud-assisted Cyber-Physical Systems

S. Sciancalepore, G. Piro, D. Caldarola, G. Boggia, and G. Bianchi

***Abstract*—While enabling brand new services and opportunities, the federation of vertical Internet of Things platforms presents new challenges in terms of secure and controlled access to heterogeneous resources, especially when authorization permissions must be regulated by multiple decentralized authorities. The work presented herein designs, develops, and experimentally validates a flexible and effective Attribute-Based Access Control framework, properly devised to operate in a federated and cloud-assisted Cyber-Physical System. Our main novelty stems in the original way we turn a policy-based encryption scheme, customarily used for accessing data, into a Cyber-Physical resource access control protocol. The proposed design approach is able to address several security issues characterizing the emerging use cases in this context, including the decoupling between authentication and authorization, fine-grained, offline, and time-limited authorization, protection against collusion attacks, access rights revocation, and user privacy. A security analysis and a performance evaluation executed through experimental tests clearly demonstrate the viability of the proposed approach in realistic cloud-assisted Cyber-Physical Systems, as well as its ability to overcome the lacks affecting competitive approaches without introducing huge communication and computational requirements.**

***Index Terms*—Cloud-assisted Cyber-Physical Systems; Federated Internet of Things; Attribute-Based Access Control**

I. Introduction

Cloud-assisted Cyber-Physical Systems (CPSs) represent a valuable approach that provides an effective interface between Internet of Things (IoT) resources and remote users, enables federation among heterogeneous platforms [1][2], and forms the basis for the sharing of resources across organizations and boundaries [3]. Its capabilities can be further extended with the key functionalities emerging from Fog computing and Mist computing architectures [4]. In this context, the protection of resources against unauthorized accesses still represents one of the main challenges to overcome. The literature clearly demonstrates that fine-grained authorization mechanisms have been easily integrated in vertical IoT platforms through single-authority and (quite often) centralized architectures [5][6]. But, federated ecosystems should inevitably integrate more complex, decentralized, and multi-authority access control schemes [3], while addressing a number of security issues (including protection against collusion attacks, time-limited authorization, simple access rights revocation, support for offline authorization, and user privacy) and embedding robust cryptographic algorithms (as suggested by the recently published General Data Protection Regulation).

The extensive literature in this field does not provide complete answers to the aforementioned challenges (see for instance [7]-[26] and the discussion reported in Sec. II). Therefore, this paper intends to extend the current state of the art by providing a twofold contribution: it (i) designs a comprehensive decentralized and multi-authority access control framework in a cloud assisted CPS, addressing all the issues listed above, and further encompassing (ii) an original usage of policy-based cryptographic schemes as a secure technique to control the access to resources, opposed to its traditional usage for data encryption/decryption.

The core of the proposed approach is a novel access control mechanism, based on the Attribute Based Access Control logic and realized through the Decentralized Multi-Authority - Ciphertext-Policy - Attribute Based Encryption (DMA-CP-ABE) algorithm [27][28]. Specifically, DMA-CP-ABE cryptographic primitives are used to implement an online authorization procedure, based on a challenge-response strategy. Moreover, during the authorization procedure, a user is identified through an ephemeral and time-limited identity, released by a trusted third-party entity in the system. In this way, the resulting scheme also supports time-limited authorization (i.e., the retrieved attributes can be used to access to resources for a limited amount of time), protection against collusion attacks (i.e., it is not possible to combine access rights belonging to different users), and the protection of user privacy (i.e., the ephemeral identity denies the resource provider to track and profile users). In addition, the aforementioned access control procedure easily supports attribute revocation through conventional approaches based on attribute revocation lists, whose security was already proven in the past.

The security of the proposed approach has been deeply investigated (see Sec. IV-A). Moreover, its performance has

S. Sciancalepore is with the College of Science and Engineering (CSE) of the Hamad Bin Khalifa University (HBKU), in Doha (Qatar); e-mail: ssciancalepore@hbku.edu.qa

G. Piro, D. Caldarola, and G. Boggia are with Dept. of Electrical and Information Engineering (DEI), Politecnico di Bari, Bari (Italy) and CNIT, Consorzio Nazionale Interuniversitario per le Telecomunicazioni; e-mail: {name.surname}@poliba.it

G. Bianchi is with Dept. of Electronic Engineering, University of Rome "Tor Vergata", Rome (Italy) and CNIT, Consorzio Nazionale Interuniversitario per le Telecomunicazioni, e-mail: {name.surname}@uniroma2.it.

This work was framed in the context of the project SymbIoTe, which receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 688156.

been experimentally evaluated in a realistic cloud-assisted CPS environment and compared against those achieved by some relevant approaches available in the literature and based on the DMA-CP-ABE algorithm. Conducted tests demonstrate that the conceived solution is able to accomplish authentication and authorization tasks in an acceptable amount of time (the maximum latency of 5.2s is registered in the more complex scenario with tens of platforms distributed worldwide and when the DMA-CP-ABE algorithm is properly configured to guarantee a high security level). At the same time, they highlight its ability to overcome the lacks of approaches available in the current state of the art, without introducing huge communication and computational requirements.

Finally, to allow both scientific and industrial communities to test, use, and extend the proposed solution, the code of developed servers, protocols, and algorithms is freely available at the link: https://telematics.poliba.it/dma-cp-abe.

The paper is structured as follows. Sec. II reviews the state of the art against the set of security issues characterizing the considered decentralized and multi-authority scenario. Sec. III presents the proposed approach. Sec. IV provides the security analysis of the conceived solution and discusses a preliminary performance assessment conducted through an experimental testbed. Finally, Sec. V draws conclusions and future directions of the work.

II. Security issues and literature review

Recent research and industrial trends are fostering the federation across vertical IoT platforms. As a result, each vertical solution is able to expose its own resources to large-scale services deployed across heterogeneous organizations and boundaries [2].

Let us now consider, for instance, a scenario embracing three different IoT platforms, that are: smart campus, smart parking, and smart restaurant. Natively, these platforms do not interact each other and a user registered in a given platform cannot access to resources exposed in another one. Thanks to the federation concept, instead, these barriers are broken: a professor registered in the smart campus domain could also obtain the access to the smart parking system or enjoy dedicated services in the smart restaurant, by only demonstrating to be registered in a platform (i.e., the smart campus) federated with the others. The resulting ecosystem assumes that [3]: (1) a user can be registered in different domains and can retrieve from each of them a wallet of properties or access grants; (2) the access to a resource is properly managed through an access control policy, which is defined as a combination of properties that can be released also outside the home domain; (3) the access to a resource can be offered to not registered users; (4) a user can obtain the access to a resource only if it is able to demonstrate to be in possession of a set of properties that jointly satisfy the aforementioned policy.

The federation of vertical IoT platforms is being also investigated by a number of initiatives funded in the European Union's Horizon 2020 research and innovation programme. Tab. I summarizes the objectives targeted by some of them, and provides a summary of security functionalities they are developing. In line with these project activities, it is possible to remark that a decentralized and multi-authority architecture requires the definition of novel security methodologies to handle robust access control schemes, while addressing the following issues:

- **Decoupling between authentication and authorization.** To provide a scalable access to resources in a federated ecosystem, authentication and authorization processes must be decoupled (i.e., executed by different components in different time instants). Each user is generally registered within a given IoT platform. During the authentication phase, it logins in that platform and receives a set of access rights. Then, these access rights can be used during the authorization procedure for gaining the access to a resource. The resulting solution is scalable because the access rights retrieved by a user can be used for performing multiple authorization processes, also in different IoT domains, until the lifetime expiration of the access rights, whose duration depends on the requirements of specific use-cases.
- **Fine-grained authorization.** The access to resources must be controlled through specific policies. A policy is formulated as a set of (potentially complex) rules that regulates the access to resources, based on the properties (i.e., access rights) possessed by clients.
- **Protection against collusion attacks.** It is necessary to bind access rights with a specific user identity. If not, malicious entities could perform collusion attacks, by mixing together properties released for different users.
- **Time limited authorization.** Access rights must have a limited lifetime, configurable according to the requirements characterizing the chosen use-case.
- **Protection of user privacy.** When accessing resources, users should expose only the minimum information necessary to successfully perform the authorization procedure. Thus, it should be hard to acquire unsolicited information and to profile any component in the ecosystem.
- **Revocation of access rights.** The system must be able to revoke, at any time, the attributes assigned to users, even before their normal expiration time.
- **Support for offline authorization.** In some use-cases, the access to the resource should be allowed even if the IoT platform is not connected to the Internet.

The scientific literature provides interesting approaches dealing with some of the aforementioned security issues, as summarized in Tab. II.

Some methodologies for conceiving privacy-by-design frameworks have been discussed in [7] and [8]. However, being rooted on privacy issues, these works do not consider other security issues faced in cloud-assisted CPS.

The contributions in [9]-[12] address authentication and authorization services in standalone IoT platforms through the OAuth 2.0 framework. Nevertheless, even if the OAuth 2.0 paradigm inherently supports the decoupling between authentication and authorization, it is not suitable for the targeted decentralized and multi-authority scenario, because it

assumes the presence of a single owner managing the access to available resources.

The outsourcing of authorization services to the cloud has been traditionally considered as a successful workaround to alleviate the computational burden incurred by standalone systems. For instance, [13] proposes a distributed architecture where IoT data are outsourced to cloud components and attribute-based authorization functionalities are handled by the cloud as well. However, authentication and authorization functionalities are handled at the same time, and the access is granted only if the entity is online.

Also in [14], IoT data are outsourced to cloud-based services and encrypted with attribute-based techniques, like Ciphertext-Policy Attribute Based Encryption (CP-ABE). Unfortunately, data outsourcing hinders the possibility to fulfill the offline authorization requirement. This problem can be also found in the flexible model proposed in [15], in which also encryption and decryption processes are outsourced to cloud services. In [16], the CP-ABE algorithm is used for offering data privacy and dynamic auditing. However, the attribute revocation is handled through the renew of the private/public key pair associated to the attribute and a particular technique, namely dual encryption. This forces all the legitimate users to renew their attributes to be still authorized in the system, resulting in a higher overhead. In addition, the access is granted once and lasts forever, without any time constraint.

The adoption of CP-ABE for access control purposes is investigated in [17]-[20]. While [17] modifies the cryptographic scheme in order to support attribute discrepancy, in [18] the CP-ABE scheme is also used for authentication of entities and resources hosted on cloud. However, in both of these approaches the access is granted only once, and lasts forever without any expiration mechanism. [19] introduces a central entity, namely Key Distribution Center, that manages access to files providing secret keys generated through the CP-ABE logic. The system, however, does not support multiple domains and authorities. In addition, the user authenticates only by using its attributes, thus the authentication and the authorization services are not decoupled. Multi-Authority CP-ABE is used in [20] to allow for the delivery of a video file to certain group of users in a determined time interval. A *Time* attribute is defined to guarantee the data freshness, while keys associated to a given attribute are renewed every time a user gets the attribute revoked. The overall approach is not scalable and it does not leverage a complete architecture able to effectively manage authentication and authorization functionalities. The simple Role Based Access Control (RBAC) logic is used in [21]-[23]. Here, fine-grained authorization is not supported because policies are primarily built on roles. User privacy is taken into account in [24] and [25]. [24] formulates a variant of the CP-ABE scheme that avoids the tracking of a user interacting with a cloud-based CPS through its Global Identifier. As in most of the previous works, the attribute revocation problem is not considered. [25] presents a solution where a user could anonymously authenticate with the infrastructure by using attribute-based tickets. The system relies on a stable communication with online components, thus being unable to fulfill the offline authorization requirement. Also, authentication and authorization services are strictly coupled, given that the user authenticates anonymously by using only its attributes. Finally, the approach presented in [26] targets efficient decryption and attribute revocation methods. However, users authenticate using their attributes and the system is not able to guarantee continuous operations even in case it is disconnected from the Internet.

As a final consideration, none of the above contributions made available the code used to implement the proposed solution, thus making very difficult the reproducibility, validation and verification of proposed results.

From the discussion above and the overview in Tab. II, it clearly emerges that a mechanism able to address, at the same time, all the considered security system issues, is still needed. Thus, the solution formulated in this work goes beyond the current state of the art because of its ability to address all the issues characterizing federated IoT ecosystem.

III. The proposed solution

The solution proposed in this work addresses all the security issues listed in Tab. II, by leveraging a novel usage of the Decentralized Multi-Authority - Ciphertext-Policy - Attribute Based Encryption algorithm.

The DMA-CP-ABE algorithm was initially conceived for offering offline data encryption, based on a given access policy, as defined in [27] and [28]. Here, an attribute is released by an authority and mapped to a set of cryptographic materials, that include: public key, private key, and secret key. The private key is kept secret by the authority. The public key is provided to the server that intends to protect the data through the DMA-CP-ABE algorithm. The secret key is calculated by jointly considering the attribute and the user identity. Moreover, it is delivered to the user after a successful authentication procedure. Once the access policy is defined, the resource provider encrypts the data by using the set of public keys related to the attributes belonging to the access policy itself. When requested, the protected data is delivered to the user. Therefore, the user processes the access policy and uses all the secret keys in its possession to decrypt the received content. In case the list of attributes retrieved in the past matches the access policy, the decryption process ends successfully. DMA-CP-ABE natively supports the decoupling between authentication and authorization. In fact, authentication and authorization procedures are separately implemented and involve different entities in the architecture. Also, the authorization process does not require that the authorities are connected to the rest of the architecture in the time instant in which the user receives the encrypted content from the server. Therefore, the algorithm can also be used in offline scenarios. The overall approach is resilient against the key escrow problem: the entity that released a sub-set of the attributes used to build the access policy cannot decrypt the protected content. The user privacy requirement is satisfied because the user is not forced to share with the server its attributes. Moreover, thanks to the possibility to create access policies with an arbitrary combination of attributes, DMA-CP-ABE also supports fine-grained authorization. Unfortunately, the

TABLE I
CURRENT TRENDS IN EUROPEAN PROJECTS.

Project	**Main goals**	**Addressed security issues**
Symbiosis of smart objects across Internet of Things environments (symbIoTe) [3]	Provides an abstraction layer for a unified control view on various IoT platforms and respective sensing/actuating resources.	Decoupling Authentication and Authorization, Fine-grained Authorization, Offline Authorization, Collusion Protection, Time-Limited Authorization, Privacy Protection, User-attribute revocation.
Bridging the Interoperability Gap in the IoT (BIG-IoT) [2]	Establishes interoperability by defining a unified Web API for IoT platforms, namely the BIG IoT API, aligned with the standards currently developed by the W3C Web of Things group.	Decoupling Authentication and Authorization, Fine-grained Authorization, Time-Limited Authorization, Privacy Protection.
INTERoperability of heterogeneous Internet of Things platforms (INTER-IoT) [29]	Facilitates the creation of an ecosystem of interoperable and open IoT platforms, with respect to the following fundamental layers: device, networking, middleware, applications, and semantics.	Fine-grained Authorization, Time-Limited Authorization, Privacy Protection, User-attribute revocation.
FEderated interoperable SmarT ICT services deVelopment And testing pLatform (FESTIVAL) [30]	Provides IoT experimentation platforms including interaction facilities with physical environments and end-users.	Decoupling Authentication and Authorization, Fine-grained Authorization, Time-Limited Authorization, Privacy Protection, User-attribute revocation.
Open virtual neighborhood network to connect Internet of Things infrastructures and smart objects (VICINITY) [6]	Create technical interoperability up to the semantic level, allowing users without technical background to get connected to the vicinity ecosystem in an easy and open way, fulfilling the consumers' needs and combining services from different domains.	Decoupling Authentication and Authorization, Privacy Protection.
TagItSmart! [31]	Creates a set of tools and enabling technologies integrated into a platform with open interfaces.	Fine-grained Authorization, Time-Limited Authorization, Privacy Protection, User-attribute revocation.
Federated Interoperable Semantic Internet of Things Testbeds and Applications (Fiesta-IoT) [32]	Provides tools, techniques, processes and best practices enabling IoT testbed/platforms operators to interconnect their facilities in an interoperable way based upon cutting edge semantics-based solutions.	Privacy Protection.

TABLE II
OVERVIEW OF THE STATE OF THE ART AGAINST THE ISSUES OF EMERGING FEDERATED IOT PLATFORMS

Security Issue	[7] - [8]	[9]-[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]	[20]	[21]	[22]	[23]	[24]	[25]	[26]	**Our Proposal**
Decoupling Authentication and Authorization		✓			✓		✓	✓				✓	✓				✓
Fine-grained Authorization			✓	✓	✓	✓	✓		✓					✓	✓	✓	✓
Offline Authorization				✓		✓			✓		✓	✓		✓			✓
Collusion Protection			✓	✓	✓	✓	✓		✓	✓	✓	✓		✓	✓	✓	✓
Time-limited Authorization					✓					✓			✓		✓		✓
Privacy Protection	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
User-attribute revocation			✓	✓	✓	✓			✓	✓		✓			✓	✓	✓
Open-Source Implementation																	✓

legacy usage of DMA-CP-ABE does not provide protection against collusion attacks. Even if the secret key is bound with the user's identity, it is possible to combine access grants given to different users and successfully decrypt a content protected with DMA-CP-ABE and an arbitrary access policy. Also, DMA-CP-ABE does not support time-limited authorization. Once the user is in possession of a proper set of attributes, it is able to decrypt contents protected with a compatible access policy forever. For the same reason, attributes revocation is not allowed.

To solve these issues, the proposed approach modifies the legacy usage of DMA-CP-ABE. The primary contribution refers to the adoption of the DMA-CP-ABE algorithm within the access control procedure. Specifically, its cryptographic primitives are used to implement a challenge-response scheme belonging to an authorization procedure, rather than encrypting the data itself. Here, the user is invited to demonstrate the possession of a subset of attributes matching the access policy protecting the requested resource (see Sec. III-E for more details). Specifically, when the user requests a resource, the contacted server encrypts a random number with DMA-CP-ABE and sends the resulting challenge to the user. Thus, the user will be able to correctly decrypt the received challenge only if it is in possession of the correct set of access rights that matches the aforementioned access policy (see Sec. III-D for more details). The output of the decryption process is delivered

to the server, which will decide to provide or not the resource according to the received answer.

In addition, the conceived solution introduces the possibility to identify the user with an ephemeral identity, that is unique within the overall cloud-assisted CPS, is released by a trusted entity, and has a limited time validity. Such an identity should be retrieved before performing authentication and authorization procedures. The ephemeral identity allows the generation of time-limited attributes during the authentication process, prevents user profiling, avoids collusion attack (see Sec. III-C for more details), and also enables the implementation of an easy revocation procedure, without incurring in additional maintenance costs (see Sec. III-F for more details).

A. Entities, functionalities, and protocol overview

The proposed solution integrates four entities, namely: **resource server**, **client application**, **attribute authority**, and **identity authority**. The resource server exposes resources and services coming from the real physical IoT world. Without loss of generality, it is possible to assume that each vertical IoT platform, having its own resources and services, uses the resource server as a stable interface for the whole cloud-assisted CPS. Indeed, the resource server also processes the requests coming from external users, verifies their authenticity, and decides to provide or deny the access to resources and services according to proper access control policies. The client application identifies the remote user willing to access to resources and services distributed across the cloud-assisted CPS ecosystem. A client application can be registered in one or more platforms. Such a registration implies the possession of a set of properties, simply referred to as *attributes*. Each single platform has an attribute authority, which is in charge of releasing trusted attributes to registered users only. In other words, the attribute authority performs user authentication and releases useful materials to be used for authorization purposes. Finally, the identity authority is the entity that assigns ephemeral identities to client applications during time.

The conceived scheme is made up of four consecutive phases, that are:

- **Phase 1: system configuration:** all the entities are configured to interact each other and perform all the activities expected for the other remaining phases. Specifically, client application, identity authority, attribute authority, and resource server are endowed with a private-public key pair, the attribute authority processes the attributes it is in charge to release according to the setup of the DMA-CP-ABE algorithm, and the resource server generates an access control policy for each resource it exposes.
- **Phase 2: ephemeral identity generation:** the client application contacts the identity authority and asks for an ephemeral identity to use during both authentication and authorization procedures. Such an ephemeral identity is uniquely associated to the real identifier of the client application through a cryptographic signed proof. The time-validity assigned to the ephemeral identity also limits the usage of attributes released by attribute authorities during the next step.
- **Phase 3: authentication procedure:** the client application logins to the attribute authorities where it is registered to and retrieves the list of attributes that encode its properties. Attributes and related cryptographic materials are then delivered to the client application through a token.
- **Phase 4: authorization procedure:** the client application sends a resource access request to the resource server. The resource server initiates the challenge-response mechanism based on the DMA-CP-ABE algorithm. This is done for recognizing the properties possessed by the client application. Also, the resource server contacts the identity authority to check if the attributes released to the client application have been revoked (note that this feature cannot be executed in offline scenarios). In case the challenge-response and the check revocation procedures are successfully completed, the access to the resource is granted. Otherwise, it is denied.

In every phase, the interaction among the involved entities is protected through the Transport Layer Security (TLS) protocol. Data confidentiality at the transport layer is indeed guaranteed.

All the technical details related to the aforementioned phases are provided in what follows.

Throughout this paper, the calligraphic uppercase letter is used to designate a set of elements, that is $\mathcal{A} = \{\ldots\}$. Instead, $\bar{a} = |\mathcal{A}|$ denotes the cardinality of the aforementioned set. The calligraphic lowercase letter, i.e., τ, is used to designate a function. Lower case letters denote a scalar. A boldface lowercase letter, i.e., $\mathbf{l}$, is used to represent a vector. The elements of a vector are listed within brackets. Therefore, in case the i-th element of a vector is identified with l_i, the vector $\mathbf{l}$ is reported as: $\mathbf{l} = [l_0, l_1, \ldots, l_{\bar{l}}]$. On the contrary, a boldface uppercase letter, i.e., $\mathbf{L}$, is used to represent a matrix. Moreover, $\mathbf{l}_i$ represents the i-th row of the matrix $\mathbf{L}$. Finally, to ease the comprehension of the notions presented in the following, a summary of the main symbols is reported in Tab. III.

B. Phase 1: system configuration

Phase 1 embraces three atomic tasks, as depicted in Fig. 1: setup of public key cryptography, definition of initial access control policies, and processing of attributes.

Each entity is endowed, during this initial phase, with a private-public key pair. In what follows, public and private keys for the client application are denoted with PU_{APP} and PK_{APP}, respectively; public and private keys for the identity authority are denoted with PU_{IA} and PK_{IA}, respectively; public and private keys for the attribute authority are denoted with PU_{AA} and PK_{AA}, respectively; and public and private keys for the resource server are denoted with PU_{RS} and PK_{RS}, respectively. The public key is stored within a trusted X.509 certificate.

The resource server configures the initial access control policies for all the resources it exposes. Let τ be the access policy defined for a given resource.

The attribute authority processes the attributes it is in charge to release according to the DMA-CP-ABE algorithm.

TABLE III
LIST OF MAIN MATHEMATICAL SYMBOLS

Symbol	Description
$\{PU_{APP}, PK_{APP}\}$	public and private key pair of a client application
$\{PU_{IA}, PK_{IA}\}$	public and private key pair of the identity authority
$\{PU_{AA}, PK_{AA}\}$	public and private key pair of an attribute authority
$\{PU_{RS}, PK_{RS}\}$	public and private key pair of a resource server
$\bar{a}$	number of attribute released by an attribute authority
$\mathcal{A}$	list of attributes released by an attribute authority
a_i	i-th attribute released by an attribute authority
$\{PU_{a_i}, PK_{a_i}\}$	public and private key pair assigned to the attribute a_i
I_{APP}	real identifier of a client application
ϵ	ephemeral identity assigned to the client application by the identity authority
a_ϵ	ephemeral attribute associated to the ephemeral identity ϵ
$\{PU_{a_\epsilon}, PK_{a_\epsilon}\}$	public and private key pair assigned to the ephemeral attribute a_ϵ
K_ϵ	secret key assigned to the ephemeral attribute a_ϵ
$K_{a_i,\epsilon}$	secret key assigned to the attribute a_i and the ephemeral identity ϵ
τ	access control policy
τ_ϵ	new access control policy that includes the ephemeral identity ϵ
$\bar{p}$	number of attributes used to create a policy
$\mathcal{P}$	list of attributes used to create a policy
p_i	i-th attribute used to create a policy
$\{PU_{p_i}, PK_{p_i}\}$	public and private key pair assigned to the attribute p_i
$\mathbf{L}$	Linear Secret Sharing Scheme matrix assigned to a policy
$\mathbf{l}_i$	i-th row if the Linear Secret Sharing Scheme matrix generated by the j-th attribute
δ	random number generated during the challenge response procedure
Ψ	cryptographic challenge generated through the DMA-CP-ABE encryption algorithm
$\bar{u}$	number of attributes possessed by the client application
$\mathcal{U}$	list of attributes possessed by the client application
u_i	i-th attribute possessed by the client application

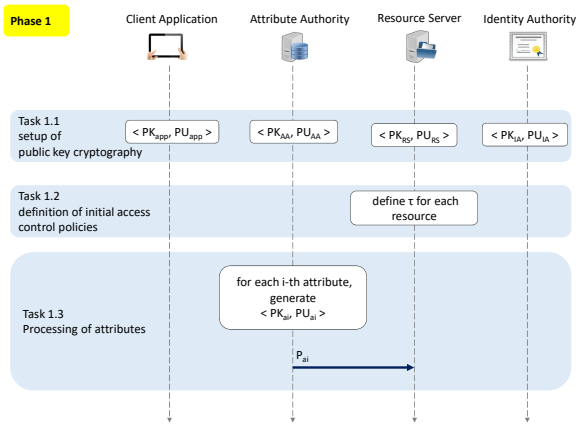


Fig. 1. Tasks executed during Phase 1 - system configuration.

It is assumed that an attribute authority is authoritative for $\bar{a} \geq 1$ different attributes, that are: $\mathcal{A} = \{a_0, a_1, \ldots, a_{\bar{a}-1}\}$. According to the DMA-CP-ABE algorithm, a unique private-public key pair is assigned to each attribute. Such keys are computed through arithmetic operations in a composite order bilinear group [27]. To this end, the generic attribute authority defines (i) a composite prime number N (product of three prime numbers) that is the order of the bilinear groups G and G_T, chosen such that the discrete logarithm problem is hard to solve on them; (ii) a generator g for the group G; (iii) a hash function H, that maps whatever string into the elements of the group G; and (iv) a bilinear map $e : G \times G \rightarrow G_T$, that is non degenerate and computable [33]. Note that the adoption of a composite prime number N as the order of the bilinear groups G and G_T allows the proposed approach to inherit the security proof of the algorithm presented in [27].

Let us consider now the $i-th$ attribute released attribute authority, namely a_i. To obtain the private and public keys (namely PK_{a_i} and PU_{a_i}, respectively) associated to the aforementioned attribute, the attribute authority firstly extracts two random exponents: $\alpha_i, \beta_i \in \mathbb{Z}$; then it computes (see [27] and [28] for more details):

$$PK_{a_i} = \{\alpha_i, \beta_i\} \qquad (1)$$
$$PU_{a_i} = \left\{e(g,g)^{\alpha_i}, g^{\beta_i}\right\}.$$

The private key is kept secret and stored by the attribute authority within a secure database. The public key, instead, is delivered to all the resource servers that use the corresponding attribute to build their access control policies.

Let K_{a_i} the secret key of the attribute. It is important to remark that it cannot be already calculated during the setup phase. It will be generated during Phase 3, by combining the private key of the attribute and the user's ephemeral identity.

C. Phase 2: ephemeral identity generation

Phase 2 is only in charge of generating an ephemeral identity and delivering it to the client application, together with its cryptography-related parameters, as illustrated in Fig. 2. Remember that the usage of an ephemeral identity extends the conventional DMA-CP-ABE scheme and makes the resulting solution able to support time-limited access grants, attribute revocation, and protection against collusion attacks.

Let I_{APP} be the real identifier of a given client application, stored in its X.509 certificate. Before starting the authentication procedure, the client application should obtain from the identity authority an ephemeral identity, ϵ. The ephemeral identity is uniquely associated to the real identifier of the client application and expires after a specific amount of time. Moreover, the relationship between the real identifier of the client application and the ephemeral identity is certified through a cryptographic mechanism. From one side, attributes released by attribute authority during the authentication phase are valid until the ephemeral identity expires. Thus, the client application must periodically renew the authentication procedure. At the same time, since attributes are directly connected to an ephemeral identity stored within a proof message, it is not possible to combine attributes assigned to different users

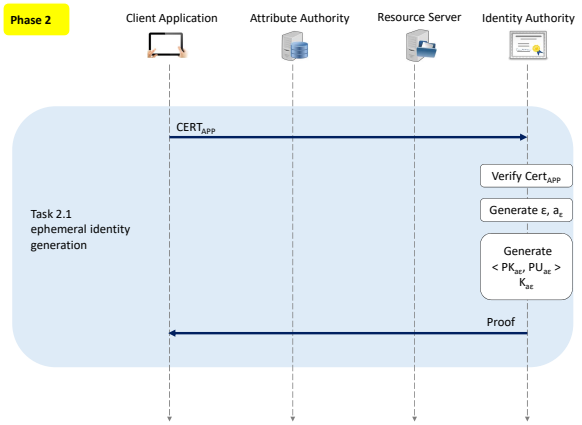


Fig. 2. Task executed during Phase 2 - ephemeral identity generation.

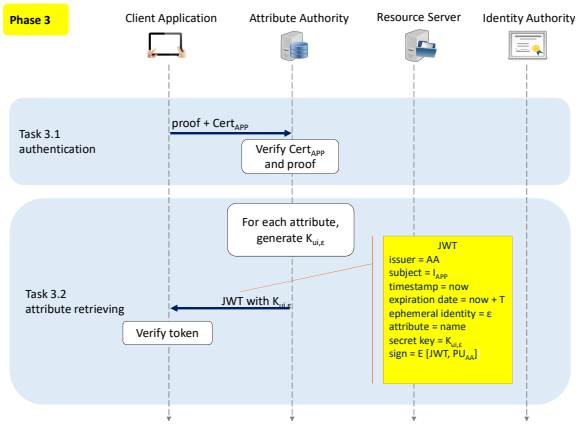


Fig. 3. Tasks executed during Phase 3 - authentication procedure.

to obtain the access to a resource. Therefore, the collusion attack is also hampered.

During Phase 2, as shown in Fig. 2, the client application initially sends its X.509 certificate to the identity authority. The identity authority verifies the authenticity of the received certificate and extracts an ephemeral identity ϵ. Then, it generates an ephemeral attribute, a_ϵ, whose private and public keys, namely PK_{a_ϵ} and PU_{a_ϵ}, respectively, are computed as (see [27] and [28] for more details):

$$\begin{aligned} PK_{a_\epsilon} &= \{\alpha_\epsilon, \beta_\epsilon\} \\ PU_{a_\epsilon} &= \left\{e(g,g)^{\alpha_\epsilon}, g^{\beta_\epsilon}\right\}, \end{aligned} \tag{2}$$

where $\alpha_\epsilon, \beta_\epsilon \in \mathbb{Z}$ are still random exponents.

Starting from the private key PK_{a_ϵ}, the ephemeral identity ϵ, and a hash function $H(\cdot)$, a secret key K_ϵ associated to the ephemeral attribute a_ϵ is obtained as (see [27] and [28] for more details):

$$K_{a_\epsilon} = g^{\alpha_\epsilon} H(\epsilon)^{\beta_\epsilon}. \tag{3}$$

The identity authority generates a proof message, which securely binds the real identity of the client application and the generated ephemeral identity. The first field of the proof is the hash function of the combination of the real identity of the client application I_{APP} and the ephemeral identity ϵ, that is $H(I_{APP}||\epsilon)$. Then, it stores the ephemeral identity ϵ, the secret key associated to the ephemeral attribute K_{a_ϵ}, the public key of the ephemeral attribute PU_{a_ϵ}, and the time validity of the ephemeral identity T. Such a message is encrypted with the private key of the identity authority, i.e., PK_{IA}:

$$proof = E_{PK_{IA}}[H(I_{APP}||\epsilon), \epsilon, K_{a_\epsilon}, PU_{a_\epsilon}, T] \tag{4}$$

Finally, the obtained proof is delivered to the client application. Phase 2 ends with the client verifying the validity of the received proof.

D. Phase 3: authentication procedure

Phase 3 embraces two atomic tasks, as depicted in Fig. 3: authentication and attribute retrieval.

As already anticipated before, each client application is registered in one ore more attribute authorities. Indeed, during the authentication procedure, the client application logins to these attribute authorities and retrieves the attributes encoding its properties. To this end, it firstly provides to the attribute authority its own credentials (e.g., username and password), the proof provided by the identity authority, and the X.509 certificate of the identity authority. Indeed, the attribute authority decrypts the proof and checks the correctness of both the corresponding time validity field and the hash of the real identity of the client application. In case the ephemeral identity is expired and/or the hash function of the real identity of the client application cannot be verified, the authentication phase ends with an error. Otherwise, the attribute authority will deliver to the client application its attributes.

According to the DMA-CP-ABE technique, an attribute is released in the form of a cryptographic material, namely *secret key*, calculated by considering the private key of the attribute (see Phase 1) and the ephemeral identity assigned by the identity authority (see Phase 2). Specifically, given the attribute u_i released by a given attribute authority, the secret key is obtained as:

$$K_{u_i,\epsilon} = g^{\alpha_i} H(\epsilon)^{\beta_i}, \tag{5}$$

where $\{\alpha_i, \beta_i\}$ and ϵ represent the private key related to the attribute u_i and the ephemeral identity, respectively.

Attributes and corresponding secret keys are delivered to the client application through standardized data structures, namely tokens. In literature, a token is frequently used as a container for security-related details, able to deliver authentication/authorization information among separated communication entities. As a matter of fact, the token is a simple means that effectively realizes the decoupling between authentication and authentication procedures. Thus, it offers the opportunity to obtain the authorization to access different resources, while performing a single authentication phase. In fact, it is released by an attribute authority after a successful authentication phase. Then, it is used in many authorization processes and

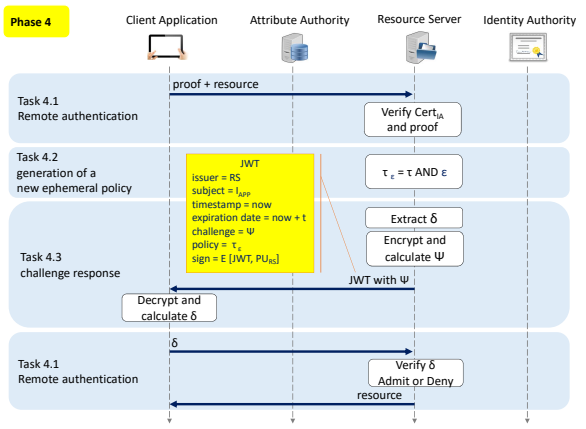


Fig. 4. Tasks executed during Phase 4 - authorization procedure.

in different and autonomous IoT domains, until the attributes stored inside are valid.

The designed approach makes use of the JSON Web Tokens technology [34]. Specifically, a JSON Web Token (JWT) uniquely binds whatever kind of information selected by the creator of the token (namely, *claims*) to the identity of the client application for which the token itself has been created. A JWT already includes few standardized claims, as issuer (i.e., who generates the token), subject (i.e., who receives the token), timestamp, and expiration date. Furthermore, additional claims are introduced: ephemeral identity, human-readable name of the attribute, and secret key. At the end of the container, the *sign* field is appended for assuring the authenticity and integrity of the token. It stores the hash function of the whole JWT, signed with the private key of the attribute authority.

From this moment on, the client application is in possession of a wallet of attributes (and the related cryptographic material) released by different attribute authorities available in different application domains. Hence, it may combine these attributes to perform the authorization procedure for each remote resource of its interest.

E. Phase 4: authorization procedure

Let us assume that the client application is interested in accessing a resource exposed by a given resource server and protected with a policy τ. The authorization procedure, depicted in Fig. 4, integrates four atomic tasks: remote authentication, generation of a new ephemeral policy, challenge response, and final service provisioning.

Once all the attributes are collected, the client application sends the request to the resource server, alongside the proof message generated by the identity authority and the X.509 certificate of the identity authority. The resource server immediately decrypts and verifies the time validity of the proof and extracts all the useful information it contains. Note that the proof does not explicitly provide information about the user (the real identifier of the client application is hidden through the hash function). Therefore, it is not possible to perform any profiling activity.

Then, the resource server creates a new ephemeral policy τ_ϵ, which combines the one originally assigned to the resource, i.e., τ, and the secret key associated to the ephemeral attribute ϵ. Practically speaking, it holds that:

$$\tau_\epsilon = \tau \text{ AND } \epsilon. \tag{6}$$

The new policy τ_ϵ is introduced in order to verify that the client application is in possession of both the required attributes (encoded in τ) and the ephemeral attribute associated to the ephemeral identity provided by the identity authority.

The client application must demonstrate to be in possession of the attributes that satisfy the access control policy through a challenge-response scheme, based on the DMA-CP-ABE algorithm. In summary, the resource server extracts a random number δ and encrypts it with DMA-CP-ABE; then, the client application decrypts the received messages and demonstrates to be able to obtain δ with the attributes in its possession.

Let $\bar{p}$, $\mathcal{P} = \{p_0, p_1, \ldots, p_{\bar{p}-1}\}$, and l be the number of attributes used by the policy τ_ϵ, the list of attributes used by the policy, and a policy-dependent parameter[1]. Moreover, $PU_{p_i} = \left\{e(g,g)^{\alpha_i}, g^{\beta_i}\right\}$ is the public key of the i-th attribute in $\mathcal{P}$, stored by the resource server.

The encryption process starts from the definition of a $\bar{a} \times l$ Linear Secret Sharing Scheme (LSSS) matrix, namely $\mathbf{L}$, which encodes the human-readable and boolean access control policy into a mathematical formulation. Specifically, a mapping function ρ_i generates the i-th row of $\mathbf{L}$, that is $\mathbf{l}_i$ starting from the attribute p_i. The interested reader can refer to [27] for details about the construction of the LSSS matrix.

The algorithm selects a random number $s \in \mathbb{Z}_N$, a random vector $\mathbf{v} \in \mathbb{Z}_N^l$, and a random vector $\mathbf{w} \in \mathbb{Z}_N^l$. Specifically, $\mathbf{v}$ is set as $\mathbf{v} = [v_0 = s, v_1, \ldots, v_{l-1}]$, having s as its first item. Moreover, $\mathbf{w}$ is set as $\mathbf{w} = [w_0 = 0, w_1, \ldots, w_{l-1}]$. Then, for each i-th attribute in $\mathcal{P}$, it extracts a random number o_i and calculates two scalar numbers, that are: $\omega_i = \mathbf{l}_i \cdot \mathbf{w}$ and $\nu_i = \mathbf{l}_i \cdot \mathbf{v}$.

Note that s, ω_i, ν_i, and PU_{p_i} are jointly used to finalize the encryption process. The resulting cryptographic material (i.e., the challenge, denoted hereafter as Ψ) is reported in Eq. (7).

$$\Psi = \begin{cases} c_0 &= \delta e(g,g)^s \\ \mathbf{c_1} &= \{c_{1,0}, c_{1,2}, \ldots, c_{1,\bar{a}-1}\} \\ \mathbf{c_2} &= \{c_{2,0}, c_{2,2}, \ldots, c_{2,\bar{a}-1}\} \\ \mathbf{c_3} &= \{c_{3,0}, c_{3,2}, \ldots, c_{3,\bar{a}-1}\} \end{cases} \tag{7}$$

where $c_{1,i} = e(g,g)^{\nu_i}\, e(g,g)^{\alpha_i o_i}$, $c_{2,i} = g^{o_i}$, and $c_{3,i} = g^{\beta_i o_i}\, g^{\omega_i}$ (see [27] for more details).

By deeply looking into Eq. (7), only the term c_0 depends on the random number δ. The other terms, instead, are related to the set of attributes belonging to the access control policy.

The resource server delivers the challenge Ψ and the policy τ_ϵ to the client application through a JWT.

After having verified the validity of the received token, the client application decrypts the received challenge. Let $\bar{u}$, $\mathcal{U} = \{u_0, u_1, \ldots, u_{\bar{u}-1}\}$ be the number of attributes and the list of attributes retrieved by the client application during the

[1]More details about the policy-dependent parameter can be found in [27].

authentication procedure, respectively. First of all, the client application checks the existence of a subset of its attributes that are compatible with the received policy. To this end, it locally generates the LSSS matrix $\mathbf{L}$ and solves the linear combination:

$$\sum_{i=0}^{\bar{u}} b_i \, \mathbf{l}_i = \{1, 0, 0, \ldots, 0\}, \tag{8}$$

where b_i are scalar coefficients $\in \mathbb{Z}_N$. Note that the $\mathbf{L}$ matrix is still calculated by means of mapping functions, as described in [27].

In the case Eq. (8) is correctly solved, the following variable is computed for each i-th attribute $u_i \in \mathcal{U}$:

$$\xi_i = \frac{c_{1,i} \, e(H(\epsilon), c_{3,i})}{e(K_{u_i,\epsilon}, c_{2,i})}. \tag{9}$$

Then, by using scalar coefficients obtained from Eq. (8), the random number δ can be computed as:

$$\delta = \frac{c_0}{\prod_{i=0}^{\bar{u}} (\xi_i)^{b_i}}. \tag{10}$$

The client application sends back to the resource server the output of Eq. (10). At this point, the resource server contacts the identity authority to verify that the attributes assigned to the client application have not been revoked (please, see Sec. III-F for more details). Indeed, the access to the resource is authorized in the case the received number is equal to the one extracted at the beginning and user's attributes have not been revoked.

F. Attribute revocation and offline authentication

One of the issues characterizing the DMA-CP-ABE algorithm is the lack of an inherent support for the revocation of access rights. Given the distributed nature of the approach and the lack of any possibility to control the use of attributes in the system, asynchronous revocation of access rights could only be provided through ancillary approaches. As already anticipated in Sec. II, available solutions use to renew the private and the public keys associated to a particular attribute when a new user is added or deleted to the list of those that possess the specific attribute [14][16][19][25]. This task is not efficient for two reasons. First, it forces all the group members to renew the considered attribute, even if it is still valid. Second, it forces the generation of a new key pair each time a user-attribute asynchronous revocation is necessary. Accordingly, very high computation, communication, and maintenance costs are required. On the contrary, the conceived solution overcomes these structural issues through an effective and cost-saving mechanism, thanks to its ability to use DMA-CP-ABE in a way that is different from the legacy one. The main advantage is given by the usage of the ephemeral identity. As discussed before, the secret key associated to a given time-limited attribute is generated by jointly considering the attribute itself and the ephemeral identity. Thus, the attribute revocation can be safely substituted with the identity revocation. Without loss of generality, it is possible to assume that the identity authority can manage an Identity Revocation List (IRL), that is a digital object containing a list of revoked ephemeral identities. As for the common X.509 Certificates Revocation Lists (CRL), the IRL is signed with any of the standard public key signature algorithms. The IRL is synchronously updated by attribute authorities and periodically updated by resource server. The IRL size cannot increase without any control, because expired ephemeral identities will be automatically deleted during time. Thus, the proposed check revocation procedure appears very lightweight and quick to execute. At the same time, it is important to remark that even if a large number of attributes are frequently renewed or revoked, no significant issues arise from the maintenance point of view. Differently from what suggested by the current state of the art, when an attribute is renewed or revoked in the proposed solution, only the secret key generated in the past is not valid anymore. Instead, no further changes are required to both private and public keys. As a consequence, the same attribute can be still used by any other user, without requiring the management of an attribute list that grows during time. As a result, the conceived solution for attribute revocation does not produce any issue related to attribute list, range of revocation, and maintenance cost.

Finally, it is worth to note that the system supports also offline access to resources. In fact, if the ephemeral identity has a suitable duration, the client application can gather the attributes when it is online and access resources hosted on a resource server that is not connected to the Internet. Obviously, in this use-case, the resource server cannot download an updated Identity Revocation List from the Identity authority and therefore the asynchronous revocation is not supported anymore. However, the resource server can report the anomaly to the system when it comes back online.

IV. Security and performance analysis

This Section provides both security analysis and performance evaluation of the conceived solution. From one side, Sec. IV-A theoretically demonstrates that the conceived approach is resilient against the most fearsome attacks. From the other side, Sec. IV-B presents a set of experimental tests, conducted for evaluating the amount of time needed to accomplish the different tasks envisaged for the designed protocol as a function of system parameters (e.g., physical distribution of nodes over the Internet and the number of attributes required to obtain the access to a given resource). The comparison against other relevant approaches, that are those presented in [24], [26], and [15], is also discussed. To make the comparison effective and fair, these approaches have been selected because they are based on DMA-CP-ABE and present a protocol structure, understood as a sequence of phases, comparable with the conceived approach (as depicted in Fig. 5) Of course, the implementation of DMA-CP-ABE and the list of tasks executed by phase are different, as anticipated in Sec. II.

A. Security analysis

Security functionalities belonging to the devised solution leverage well-known building blocks (like TLS, DMA CP-ABE, and JWT technologies). These blocks remain independent in their construction and their security has been

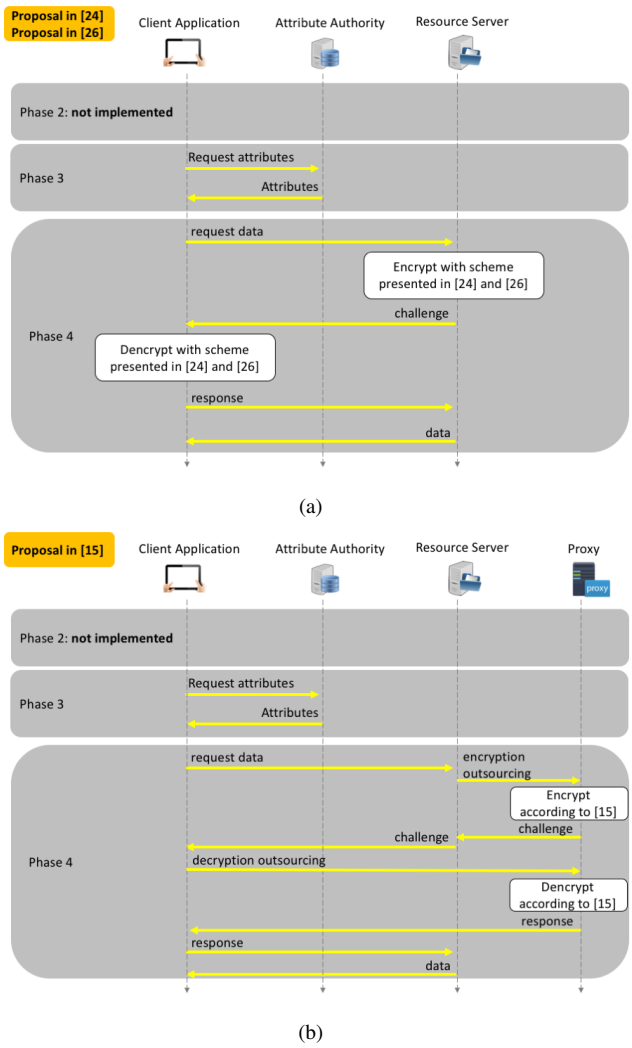


Fig. 5. High-level description of approaches proposed in [24], [26] and [15] and taken into account for the performance evaluation.

already proved in the past and formally presented in reference contributions cited below. In this way, we avoid possible issues that may emerge when mixing blocks whose conjunct adoption is not universally guaranteed ([35], for instance, shows problems related to the composition of implicit certificates and ECDSA technique). Indeed, the security analysis of each atomic security functionality is reported below:

- **Entity authentication.** During the configuration phase (see Sec. III-B for more details), each entity is endowed with a private-public key pair. Moreover, the public key is also stored within a X.509 certificate. The security of X.509 certificates is related to the cryptography technique used to generate the digital sign. Rivest-Shamir-Adleman (RSA) and Elliptic Curve Digital Signature Algorithm (ECDSA) algorithms can be used for this purpose. Their security was proved in [36] and [37].
- **Secure end-to-end channel established through TLS.** The envisaged architecture assumes to establish a secure communication between any node pair, based on the TLS protocol. It is used to offer both data confidentiality and peer authentication, while making the system resilient against Man-In-The-Middle (MITM) attacks. TLS is a well-known and widely used security protocol. Its security proof was already discussed in [38] and [39].
- **Cryptographic operations related to DMA-CP-ABE.** The proposed solution embeds cryptographic attributes generated through the DMA-CP-ABE technique. Also in this case, the security of this cryptographic building block was proved in [27]. Differently from the original version of the algorithm, in this work the secret keys associated to attributes are generated starting from an ephemeral identity. Nevertheless, the security level of the algorithm is not influenced because it is independent from the particular value assigned to the user identity. In addition, DMA-CP-ABE can suffer from attacks related to the implementation of elliptic curve and pairing based cryptography operations, such as fault and side-channel attacks [40]. Efficient protection against these kinds of attacks can be guaranteed by carefully selecting the underlying elliptic curve for cryptography operations and by adopting specific countermeasures at the hardware side (as the Montgomery ladder algorithm, used for instance in [41]).
- **Identity collusion.** When multiple properties can be retrieved from independent domains, the collusion attack may happen: the client application may perform the authentication process on behalf of different users and then combine obtained attributes for being authorized to access to a given resource. The procedure defined for both Phase 1 and Phase 2, discussed in Sec. III-C and Sec. III-D, respectively, makes the proposed solution resilient against the collusion attack. Trusted attributes, in fact, are associated through a cryptographic proof to a single user, that is in possession of a unique X.509 certificate. When an ephemeral identity is assigned to the user, it is still associated to the real identity stored within the X.509 certificate. Therefore, identity collusion cannot be achieved (i.e., two or more users cannot combine their attributes to obtain the access to a given resource).
- **Security of tokens.** With reference to Phase 3 and Phase 4, discussed in Sec. III-D and Sec. III-E, respectively, the cryptographic material is exchanged through a standardized data structure, that is the JWT token. The security of JWT tokens depends on the public-key cryptography technique used to generate the digital sign. Also in this case, RSA and ECDSA algorithms can be used for this purpose. Their security was proved in [36] and [37].
- **User privacy.** Furthermore, the proposed approach provides inherent benefits in terms of privacy. In fact, attributes possessed by a client application are never exposed to any party. Accordingly, the risk of information exposure is minimized. At the same time, the risk of activity *tracking* by third-parties is avoided because the ephemeral identifier of the client application changes periodically and its relationship with the real user identity is cryptographically protected.
- **Attribute and identity revocation.** The revocation of user's identity and related attributes is simply implemented through a Identity Revocation List (IRL), man-

aged by the identity authority. The security of a general revocation list, including the aforementioned IRL, has been already proved in the past and also standardized by the most important authorities in the security field [42][43].

B. Performance assessment

The performance assessment discussed herein considers a realistic[2] cloud-assisted CPS made up of one client application, N different attribute authorities, one identity authority, and one resource server. While the client application is physically located in the south of Italy (specifically, at the Telematics Laboratory of the Polytechnic University of Bari), the other entities are worldwide distributed on clouds. The scenario is depicted in Fig. 6. Without loss of generality, it is assumed that the client application is registered in such N attribute authorities. Therefore, it may perform authentication procedures (as illustrated in Sec. III-C) with all of them, thus being able to retrieve trusted attributes belonging to N different IoT domains. To simplify, it is also supposed that each attribute authority is authoritative for one attribute only. The resource server exposes a resource, whose access policy is properly configured by taking into account the set of attributes that the aforementioned attribute authorities may release. Thus, the client application is able to obtain the access to such a resource only if it demonstrates to have successfully finalized the N parallel authentication procedures (e.g., see the Phase 3 depicted in Fig. 3, executed for each attribute authority where the client application is registered to).

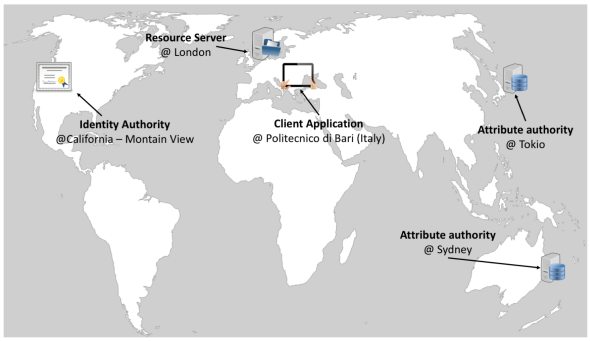


Fig. 6. An example of a cloud-assisted CPS ecosystem.

To evaluate the performance of the proposed approach, two kinds of tests have been conducted. The former aims at measuring the amount of time required to execute each single phase characterizing the designed solution. The latter, instead, evaluates the aggregated delay needed to complete the whole authentication and authorization process as a function of the number of attributes used to create the access policy. Anyway, the presented study only considers the impact of the ephemeral identity generation, the authentication procedure, and the authorization procedure. Since Phase 1 is devoted to the system configuration and it is executed only at the beginning of the system deployment, it has been not considered, given that it does not introduce any additional delay during authentication and authorization procedures. Each test was executed 200 times. Moreover, all the achieved results have been processed to show both the average value and the 95% confidence interval, estimated with the Gauss statistics.

The security level of the proposed approach is directly associated with the length of cryptographic keys involved in security operations. From one side, the longer the keys, the higher the achieved security level. From another side, the longer the keys, the higher the amount of time necessary to complete the whole protocol. Therefore, in order to provide a clear idea about the required computational costs and to give the opportunity to properly choose the configuration that offers a good compromise between security level and communication latencies, the tests have been also conducted by considering two different sizes of the base field of the elliptic curve used to perform pairing operations, i.e., 256 bits and 1536 bits. The elliptic curve presented in [44] is used for handling Elliptic Curve Cryptography (ECC) operations. Moreover, the size of the base field of the elliptic curve adopted in experimental tests is able to guarantee the minimum acceptable security level (i.e., equal to 80) [45].

1) The experimental testbed: From the implementation point of view, identity authority, attribute authorities, and resource server have been developed by using the Django python framework. Furthermore, their related databases have been managed with PostgreSQL. The open-source and freely-available python implementation of DMA-CP-ABE[3] was properly extended in order to implement the functionalities of the devised solution. The python *JWT.io* library was adopted to create and manage tokens. Finally, the client application was developed as a web-based interactive tool. Identity authority, attribute authorities, and resource server have been installed in a workstation with Ubuntu 16.10 operating system, Intel(R) Core (TM) i5-6400 CPU working at 2.70GHz, and 8GB of RAM. The client application is implemented as a web-based application and installed in a laptop directly connected to the aforementioned workstation.

It is important to note that even if all the communicating entities are physically installed within only two machines directly connected each other, the realistic distribution of identity authority, attribute authorities, and resource server is modeled at the communication level by introducing additional latencies in each interaction involving any node pair. Specifically, these latencies are statistically modeled by considering end-to-end communication delays measured between two real end points attached to Internet. To this end, ten different servers have been identified. Then, a train of *ICMP echo requests* have been sent from a laptop that hosts the client application (i.e., the node deployed at the Telematics Laboratory of the Polytechnic University of Bari) to the aforementioned well-known servers.

[2]See https://aws.amazon.com/en/about-aws/global-infrastructure/ for a direct comparison with the cloud infrastructure of AWS.

[3]http://jhuisi.github.io/charm/charm/schemes/dabeaw11.html?highlight=abe#dabeaw11

Finally, the *RTT* values have been measured as the difference between the time instant in which the *ICMP echo request* is sent and the time instant in which the corresponding answer (i.e., the *ICMP echo reply*) is received. Obtained results are reported in Fig. 7. The Round Trip Time (RTT) samples have been statistically processed in order to obtain empirical Cumulative Distribution Functions (CDFs), as illustrated in Fig. 8. These curves have been used to randomly calculate the communication delays incurring between any node pair. Specifically, in each experiment, the position of a cloud entity is randomly chosen among those reported in Fig. 7. Therefore, when the client application has to exchange data with another entity in the cloud, a random delay extracted from the RTT curve associated to its position is added to the communication process.

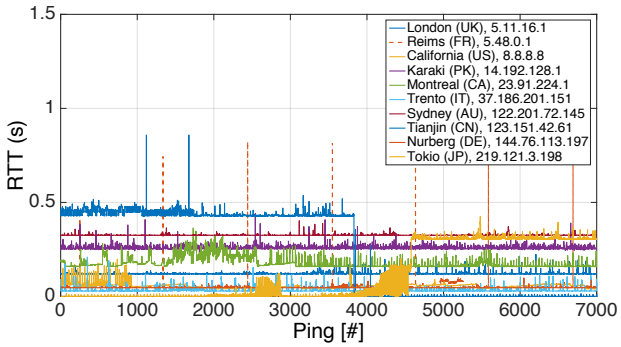


Fig. 7. RTT values measured for ten different servers deployed worldwide.

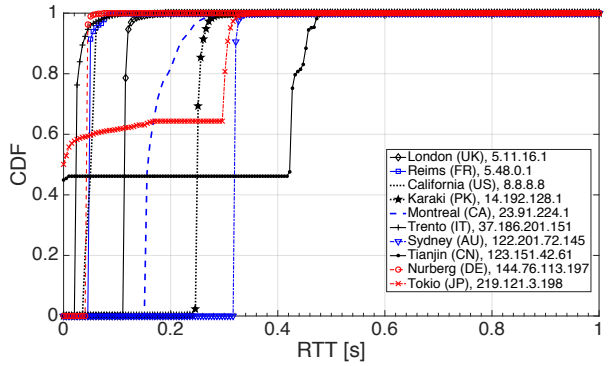
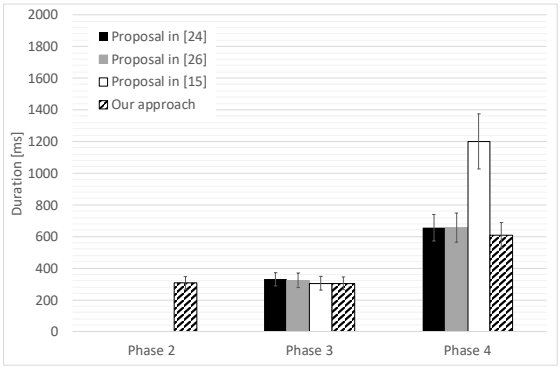
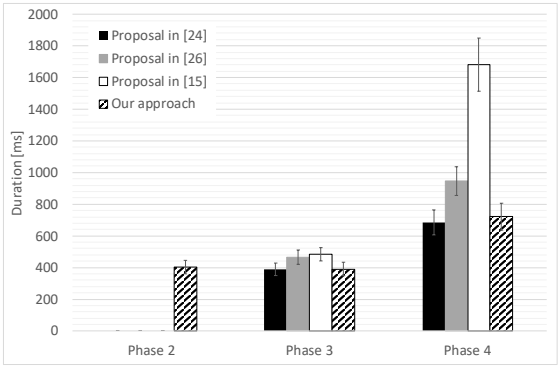


Fig. 8. Resulting empirical CDFs of communication delays in a realistic cloud-assisted CPS, against different well-known servers deployed worldwide.

2) Obtained results: Fig. 9(a) and Fig. 9(b) report the amount of time needed to complete the latest three phases of the protocol, when the size of the base field of the elliptic curve is set to 256 bits and 1536 bits, respectively. With reference to the proposed solution, Phase 2 and Phase 3 are approximately completed within the same amount of time, that falls in the range [0.3, 0.4] s. Phase 4, instead, appears as the most time consuming part of the protocol, and it takes more than 0.6 s. Differently from Phase 2 and Phase 3, in fact, Phase 4 implements complex cryptographic operations. Moreover, Phase 4 requires the exchange of a number of messages that is twice the previous phases. The other solutions taken into account for the comparison, do not implement tasks envisaged for Phase 2. Accordingly, in this case they register no latencies. Focusing on Phase 3, it is possible to observe that similar delays are achieved by all the approaches when the size of the base of the elliptic curve is set to 256 bits. But, in case the base of the elliptic curve is set to 1536 bits, the solutions available in the literature register higher delays, due to the heavier implementation of DMA-CP-ABE cryptography primitives. Regarding Phase 4, instead, it clearly emerges how the solution proposed in [15] always reaches higher latencies. With this approach, in fact, encryption and decryption operations are outsourced to a trusted proxy, thus requiring four additional messages to be exchanged and a consequent increase in the communication delays.



(a)



(b)

Fig. 9. Amount of time needed to perform the latest three phases of the protocol, when the size of the base of the elliptic curve is set to (a) 256 bits and (b) 1536 bits.

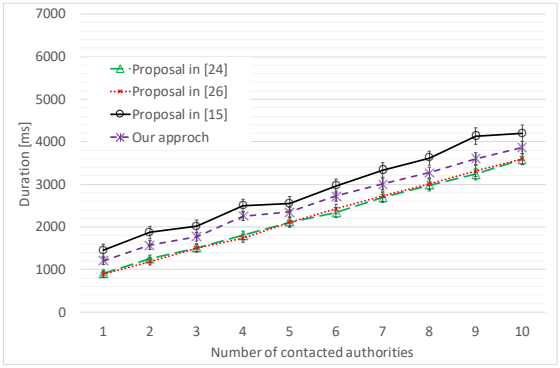
Fig. 10(a) and Fig. 10(b) show the aggregate amount of time needed to finalize the entire authentication and authorization procedure as a function of the number of attributes forming the access policy, when the size of the base of the elliptic curve is set to 256 bits and 1536 bits, respectively. As expected, the latencies increase with the number of attributes (or, in other words, with the number of attribute authorities). This result can be justified by considering that, for each attribute

in the access policy, the client application has to contact a dedicated attribute authority. Therefore, the total amount of time necessary to perform the authentication procedure increases with the number of attribute authorities. Furthermore, the obtained results confirm the impact of the key size to the amount of time needed to complete the overall protocol. When the base field of the elliptic curve is set to 256 bits, the security architectures proposed in [24] and [26] guarantee the access to a resource in less amount of time than the proposed approach. When the base field of the elliptic curve is set to 1536 bits, instead, only the solution presented in [24] registers lower latencies. In general, these results are due to the fact that these schemes presented in [24] and [26] do not implement any task of Phase 2, dedicated to the interaction with the identity authority. But, in contrast, they are not able to address time-limited authorization and user privacy issues. The security architecture proposed in [15] represents the most time-consuming, because of the outsourcing of encryption and decryption operations to the remote and trusted proxy, and it is not suitable for offline scenarios (as discussed in Sec. II).

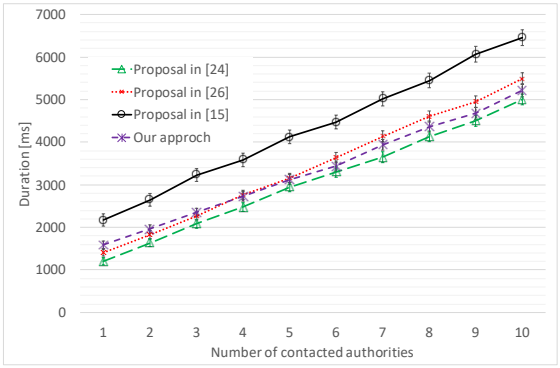
It is possible to conclude that the above comparison clearly demonstrates that the proposed solution guarantees the best compromise between the fulfillment of all the requirements (including time-limited authorization, user privacy, and off-line scenarios) and the amount of time needed to implement the overall protocol.

V. Conclusion and Future Works

This work formulated a novel methodology for handling authentication and authorization procedures in large scale cloud-assisted Cyber-Physical Systems, encompassing access control mechanisms in decentralized and multi-authority scenarios. To efficiently enable attribute-based access control in the presence of multiple, independent, and federated platforms, the conceived solution leverages and properly extends the well-known DMA-CP-ABE algorithm. In addition, it jointly addresses a number of security issues, including decoupling between authentication and authorization, mutual authentication, support for offline authorization, protection against collusion attacks, time-limited authorization, access rights revocation, and user privacy. All the security functionalities belonging to the proposed approach are based on conventional building blocks, whose security was already proved in the past. Since these blocks remains independent in their construction, the security of the whole solution is still guaranteed. Experimental tests have been conducted to measure the time required to accomplish authentication and authorization services in different system configurations. Obtained results demonstrated that communication latencies are not too high, always less than 5.5 seconds, even if complex tasks and many worldwide interactions among peers are required. Future directions of this work include the comparison of the framework with other access control architectures that are currently under design and development in the major international projects and the evaluation of system performances in the presence of real applications.



(a)



(b)

Fig. 10. Amount of time needed to complete the whole protocol as a function of the number of attributes used to create the access policy, when the size of the base of the elliptic curve is set to (a) 256 bits and (b) 1536 bits.

References

[1] M. A. Razzaque, M. Milojevic-Jevric, A. Palade, and S. Clarke, "Middleware for Internet of Things: A Survey," *IEEE Internet of Things Journal*, vol. 3, no. 1, pp. 70–95, Feb. 2016.

[2] A. Bröring, S. Schmid, C. K. Schindhelm, A. Khelil, S. Kbisch, D. Kramer, D. L. Phuoc, J. Mitic, D. Anicic, and E. Teniente, "Enabling IoT Ecosystems through Platform Interoperability," *IEEE Software*, vol. 34, no. 1, pp. 54–61, Jan. 2017.

[3] S. Sciancalepore, M. Pilc, S. Schröder, G. Bianchi, G. Boggia, M. Pawłowski, G. Piro, M. Płóciennik, and H. Weisgrab, "Attribute-Based Access Control scheme in federated IoT platforms," in *Proc. of 2nd Workshop on Interoperability and Open-Source Solutions for the Internet of Things*, ser. LCNS. Stuttgart, Germany: Springer, Nov. 2016.

[4] K. Yeow, A. Gani, R. W. Ahmad, J. J. P. C. Rodrigues, and K. Ko, "Decentralized Consensus for Edge-Centric Internet of Things: A Review, Taxonomy, and Research Issues," *IEEE Access*, vol. 6, pp. 1513–1524, 2018.

[5] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A Survey on Security and Privacy Issues in Internet-of-Things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1250–1258, Oct. 2017.

[6] Y. Guan, J. C. Vasquez, J. M. Guerrero, N. Samovich, S. Vanya, V. Oravec, R. Garca-Castro, F. Serena, M. Poveda-Villaln, C. Radojicic, C. Heinz, C. Grimm, A. Tryferidis, D. Tzovaras, K. Dickerson, M. Paralic, M. Skokan, and T. Sabol, "An open virtual neighbourhood network to connect IoT infrastructures and smart objects — Vicinity: IoT enables interoperability as a service," in *2017 Global Internet of Things Summit (GIoTS)*, Jun. 2017, pp. 1–6.

[7] P. Schaar, "Privacy by design," *Identity in the Information Society*, vol. 3, no. 2, pp. 267–274, Aug 2010.

[8] A. Cavoukian, "Privacy by design [leading edge]," *IEEE Technology and Society Magazine*, vol. 31, no. 4, pp. 18–19, 2012.

[9] S. Cirani, M. Picone, P. Gonizzi, L. Veltri, and G. Ferrari, "IoT-OAS: An OAuth-Based Authorization Service Architecture for Secure Services in IoT Scenarios," *IEEE Sensors J.*, vol. 15, no. 2, pp. 1224–1234, Feb. 2015.

[10] C. Pisa, A. Caponi, T. Dargahi, G. Bianchi, and N. Blefari-Melazzi, "WI-FAB: Attribute-based WLAN Access Control, Without Pre-shared Keys and Backend Infrastructures," in *Proc. of the ACM Int. Worksh. on Hot Topics in Planet-scale mObile Computing and Online Social neTworking*, 2016, pp. 31–36.

[11] S. Sciancalepore, G. Piro, D. Caldarola, G. Boggia, and G. Bianchi, "OAuth-IoT: An access control framework for the Internet of Things based on open standards," in *2017 IEEE Symposium on Computers and Communications (ISCC)*, Jul. 2017, pp. 676–681.

[12] A. Tassanaviboon and G. Gong, "OAuth and ABE Based Authorization in Semi-trusted Cloud Computing: Aauth," in *Proceedings of the Second International Workshop on Data Intensive Computing in the Clouds*, 2011, pp. 41–50.

[13] D. Ramesh and R. Priya, "Multi-authority scheme based CP-ABE with attribute revocation for cloud data storage," in *Int. Conf. on Microelectronics, Computing and Communications*, Jan. 2016, pp. 1–4.

[14] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131–143, Jan. 2013.

[15] S. Zhou, R. Du, J. Chen, J. Shen, H. Deng, and H. Zhang, "FACOR: Flexible access control with outsourceable revocation in mobile clouds," *China Communications*, vol. 13, no. 4, pp. 136–150, Apr. 2016.

[16] L. Y. Yeh, P. Y. Chiang, Y. L. Tsai, and J. L. Huang, "Cloud-based Fine-grained Health Information Access Control Framework for Lightweight IoT Devices with Dynamic Auditing and Attribute Revocation," *IEEE Transactions on Cloud Computing*, vol. PP, no. 99, pp. 1–1, Oct. 2015.

[17] S. Zhu and G. Gong, "Fuzzy Authorization for Cloud Storage," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 422–435, Oct. 2014.

[18] L. Wang, Q. Xie, and H. Zhong, "Cooperative Query Answer Authentication Scheme Over Anonymous Sensing Data," *IEEE Access*, vol. 5, pp. 3216–3227, Mar. 2017.

[19] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed Access Control in Clouds," in *Int. Conf. on Trust, Security and Privacy in Computing and Communications*, Nov. 2011, pp. 91–98.

[20] K. Yang, Z. Liu, X. Jia, and X. S. Shen, "Time-Domain Attribute-Based Access Control for Cloud-Based Video Content Sharing: A Cryptographic Approach," *IEEE Transactions on Multimedia*, vol. 18, no. 5, pp. 940–950, May 2016.

[21] Y. Zhu, D. Huang, C. J. Hu, and X. Wang, "From RBAC to ABAC: Constructing Flexible Data Access Control for Cloud Storage Services," *IEEE Transactions on Services Computing*, vol. 8, no. 4, pp. 601–616, Jul. 2015.

[22] B. Lang, J. Wang, and Y. Liu, "Achieving Flexible and Self-Contained Data Protection in Cloud Computing," *IEEE Access*, vol. 5, pp. 1510–1523, Feb. 2017.

[23] J. M. A. Calero, N. Edwards, J. Kirschnick, L. Wilcock, and M. Wray, "Toward a Multi-Tenancy Authorization System for Cloud Services," *IEEE Security Privacy*, vol. 8, no. 6, pp. 48–55, Nov. 2010.

[24] J. Chen and H. Ma, "Privacy-Preserving Decentralized Access Control for Cloud Storage Systems," in *Int. Conf. on Cloud Comput.*, Jun. 2014, pp. 506–513.

[25] C. Buttner and S. A. Huss, "Attribute-based authorization tickets for Car-to-X communication," in *IEEE Conf. on Communications and Network Security (CNS)*, Oct. 2016, pp. 234–242.

[26] K. Yang, X. Jia, K. Ren, B. Zhang, and R. Xie, "DAC-MACS: Effective Data Access Control for Multiauthority Cloud Storage Systems," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 11, pp. 1790–1801, Nov. 2013.

[27] A. Lewko and B. Waters, "Decentralizing Attribute-based Encryption," in *Proc. of the 30th Annual Int. Conf. on Theory and Applications of Cryptographic Techniques: Advances in Cryptology*, 2011, pp. 568–588.

[28] A. Lewko and B. Waters, "New proof methods for attribute-based encryption: Achieving full security through selective techniques," in *in Proc. of CRYPTO*, 2012, pp. 180–198.

[29] G. Fortino, C. Savaglio, C. E. Palau, J. S. de Puga, M. Ganzha, M. Paprzycki, M. Montesinos, A. Liotta, and M. Llop, *Towards Multi-layer Interoperability of Heterogeneous IoT Platforms: The INTER-IoT Approach*. Integration, Interconnection, and Interoperability of IoT Systems, Springer, 2018, pp. 199–232.

[30] T. Akiyama, S. Murata, K. Tsuchiya, T. Yokoyama, M. Maggio, G. Ciulla, J. R. Santana, M. Zhao, J. B. D. Nascimento, and L. Grgen, "FESTIVAL: Design and Implementation of Federated Interoperable Smart ICT Services Development and Testing Platform," *Journal of Information Processing*, vol. 25, pp. 278–287, 2017.

[31] S. andi, S. Radonji, J. Drobnjak, M. Simeunovi, B. Stamatovi, and T. Popovi, "Smart tags for brand protection and anti-counterfeiting in wine industry," in *2018 23rd International Scientific-Professional Conference on Information Technology (IT)*, Feb. 2018, pp. 1–5.

[32] M. Serrano, A. Gyrard, E. Tragos, and H. Nguyen, "FIESTAIoT Project: Federated Interoperable Semantic IoT/Cloud Testbeds and Applications," in *Companion Proceedings of the The Web Conference 2018*, 2018, pp. 425–426.

[33] T. Okamoto, "Cryptography Based on Bilinear Maps," in *International Symposium on Applied Algebra, Algebraic Algorithms and Error-Correcting Code*, Feb. 2006, pp. 35–50.

[34] *JSON Web Token (JWT)*, Internet Engineering Task Force Std. RFC7519, 2015.

[35] D. R. Brown, M. J. Campagna, and S. A. Vanstone, "Security of ECQV-Certified ECDSA Against Passive Adversaries," *IACR Cryptology ePrint Archive*, 2009.

[36] "PKCS 1 v2.1: RSA Cryptography Standard," 2002.

[37] I. Blake, G. Seroussi, and N. Smart, *Elliptic Curves in Cryptography*. Cambridge University Press, 1999.

[38] A. K. Ranjan, V. Kumar, and M. Hussain, "Security analysis of TLS authentication," in *Proc. of Int. Conf. on Contemporary Computing and Informatics (IC3I)*, Nov. 2014, pp. 1356–1360.

[39] A. Ferreira, R. Giustolisi, J. L. Huynen, V. Koenig, and G. Lenzini, "Studies in Socio-technical Security Analysis: Authentication of Identities with TLS Certificates," in *Proc. of IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications (TrustCom)*, July 2013, pp. 1553–1558.

[40] N. El Mrabet, J. J. Fournier, L. Goubin, and R. Lashermes, "A Survey of Fault Attacks in Pairing Based Cryptography," *Cryptography Commun.*, vol. 7, no. 1, pp. 185–205, Mar. 2015.

[41] S. Sciancalepore, G. Piro, G. Boggia, and G. Bianchi, "Public Key Authentication and Key Agreement in IoT Devices With Minimal Airtime Consumption," *IEEE Embedded Systems Letters*, vol. 9, no. 1, pp. 1–4, Mar. 2017.

[42] "ISO/IEC/IEEE International Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Part 1AR: Secure device identity," *ISO/IEC/IEEE 8802-1AR:2014(E)*, pp. 1–82, Feb. 2014.

[43] M. Naor and K. Nissim, "Certificate revocation and certificate update," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 4, pp. 561–570, Apr. 2000.

[44] A. De Caro and V. Iovino, "jpbc: Java pairing based cryptography," in *IEEE Proc. of Symposium on Computers and Communications (ISCC)*. Kerkyra, Corfu, Greece, June 28 - July 1: IEEE, 2011, pp. 850–855.

[45] ECRYPT, *Yearly Report on Algorithms and Keysizes*, 2015.

Savio Sciancalepore is currently Post Doc Researcher at the College of Science and Engineering of the Hamad Bin Khalifa University (HBKU), Doha, Qatar. He received his master degree in Telecommunications Engineering in 2013 and his PhD in 2017 in Electric and Information Engineering, both from the Politecnico di Bari, Italy. He received the prestigious award from the ERCIM Security Trust and Management (STM) Working Group for the best Ph.D. Thesis in Information and Network Security in 2018. From 2016 to 2017, he was involved in the EU H2020 project symbIoTe. His major research interests include security issues in Internet of Things (IoT) and wireless networks, as well as wireless localization techniques.

Giuseppe Piro (S'10-M'13) is an Assistant Professor at "Politecnico di Bari", Italy. He received a first level degree and a second level degree (both cum laude) in Telecommunications Engineering from "Politecnico di Bari", Italy, in 2006 and 2008, respectively. He received the Ph.D. degree in Electronic Engineering from "Politecnico di Bari", Italy, on March 2012. His main research interests include quality of service in wireless networks, network simulation tools, 5G cellular systems, Information Centric Networking, nano communications, and Internet of Things. He founded LTE-Sim and NANO-SIM projects. His research activity is documented more than 70 peer-reviewed international journals and conference papers, accounting for more than 2500 citations and a H-index of 19 (Scholar Google). He is regularly involved as member of the TPC of many prestigious international conferences. Currently, he serves as Associate Editor for Internet Technology Letter (Wiley), Wireless Communications and Mobile Computing (Hindawi), and Sensors (MDPI).

Daniele Caldarola is an MSc Student at "Politecnico di Bari", Italy. He received a first level degree in Electronic Engineering from "Politecnico di Bari", Italy, in 2011. He worked as a systems analyst and researcher for Dyrecta Lab, Research Institute, Italy, in 2014-15, privileging computer vision algorithms and embedded systems integration. In 2016, he started working as a researcher at CNIT (Consorzio Nazionale Interuniversitario per le Telecomunicazioni), Italy, for the H2020 SymbIoTe project focusing on the development of secured web services in distributed IoT platforms. His main academic interests include wireless sensor networks, cryptography, embedded electronics, system integration and analysis.

Gennaro Boggia (S'99-M'01-SM'09) received, with honors, the Dr. Eng. and Ph.D. degrees in electronics engineering, both from the Politecnico di Bari, Bari, Italy, in July 1997 and March 2001, respectively. Since September 2002, he has been with the Department of Electrical and Information Engineering, Politecnico di Bari, where he is currently an Associate Professor. From May 1999 to December 1999, he was a Visiting Researcher with the TILab, TelecomItalia Lab, Torino, Italy, where he was involved in the study of the core network for the evolution of Third-Generation (3G) cellular systems. In 2007, he was a Visiting Researcher at FTW, Vienna, Austria, where he was involved in activities on passive and active traffic monitoring in 3G networks. He has authored or coauthored more than 150 papers in international journals or conference proceedings, gaining more than 2300 citations. He is active in the IETF ICNRG working group and in the IEEE WG 6TiSCH. He is also regularly involved as a Member of the Technical Program Committee of many prestigious international conferences. His research interests include the fields of Wireless Networking, Cellular Communication, Internet of Things (IoT), Network Security, Security in Iot, Information Centric Networking (ICN), Protocol stacks for industrial applications, Internet measurements, and Network Performance Evaluation. Dr. Boggia is currently an Associate Editor for the IEEE Commun. Mag., the ETT Wiley Journal, and the Springer Wireless Networks journal.

Giuseppe Bianchi is Full Professor of Networking at the School of Engineering, University of Roma Tor Vergata since January 2007. His research activity includes programmable network systems, wireless networks, privacy and security, traffic control, and is documented in about 220 peer-reviewed international journal and conference papers, having received more than 15.000 citations (source scholar.google.com). He has been general or technical co-chair for several major conferences and workshops (IEEE INFOCOM 2014, ACM CoNext 2015, IEEE LANMAN 2016, IEEE HPSR 2018, ITC 2018, etc), and has held general or technical coordination roles in several European projects (FP6-DISCREET, FP7-FLAVIA, FP7-PRISM, FP7-DEMONS, H2020-BEBA, H2020-SCISSOR) on wireless, network programmability, 5G, and network security topics. He has been (or still is) editor for several journals in his field, including IEEE/ACM Transactions on Networking, IEEE Transactions on Wireless Communications, IEEE Transactions on Network and Service Management, and Elsevier Computer Communications.