IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 11, 2019, accepted July 1, 2019, date of publication July 3, 2019, date of current version July 19, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2926556

An Identity Framework for Providing Access to FIWARE OAuth 2.0-Based Services According to the eIDAS European Regulation

ÁLVARO ALONSO[1], ALEJANDRO POZO[2], JOHNNY CHOQUE[3], GLORIA BUENO[4], JOAQUÍN SALVACHÚA[2], LUIS DIEZ[3], JORGE MARÍN[4], AND PEDRO LUIS CHAS ALONSO[2]

[1]Departamento de Ingeniería Telemática y Electrónica, Universidad Politécnica de Madrid, 28040 Madrid, Spain
[2]Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, 28040 Madrid, Spain
[3]Departamento de Ingeniería de Comunicaciones, Universidad de Cantabria, 39005 Santander, Spain
[4]MashmeTV, 28037 Madrid, Spain

Corresponding author: Álvaro Alonso (aalonsog@dit.upm.es)

This work was supported by the CEF Telecom under the action CEF eID-FIWARE with Grant 2017-ES-IA-0042.

ABSTRACT Secure electronic identification (eID) is one of the key enablers of data protection, privacy, and the prevention of online fraud. However, until now, the lack of common legal basis prevented European Member States from recognizing and accepting eIDs issued in the other Member States. The electronic identification and trust services (eIDAS) regulation provides a solution to these issues by ensuring the cross-border mutual recognition of eIDs. FIWARE is a European initiative that provides a rather simple yet powerful set of application programming interfaces (APIs) that ease the development of smart applications in multiple vertical sectors and oriented to the future internet. In this paper, we propose a model that enables the connection of FIWARE OAuth 2.0-based services with the eID authentication provided by eIDAS reference. Thanks to this model, services already connected with an OAuth 2.0 identity provider can be automatically connected with eIDAS nodes for providing eID authentication to European citizens. For validating the proposed model, we have deployed an instance of the FIWARE identity manager connected to the Spanish eIDAS node. Then, we have registered two services, a private videoconferencing system, and a public smart city deployment, and extended their functionalities for enriching the user experience leveraging the eID authentication. We have evaluated the integration of both services in the eIDAS network with real users from seven different countries. We conclude that the proposed model facilitates the integration of generic and FIWARE-based OAuth 2.0 services to the eIDAS infrastructure, making the connection transparent for developers.

INDEX TERMS Access Control, eIDAS, electronic identification, identity, FIWARE.

I. INTRODUCTION

Secure electronic identification (eID) is one of the key enablers of data protection, privacy and the prevention of online fraud, especially in new areas of application, like Smart Cities, where incorporating real identities into trustable infrastructures has a huge potential.

eID can guarantee the unambiguous identification of a person and make it possible to get the service delivered to the person who is really entitled to it. However, until now, the lack of common legal basis prevented European Member States from recognizing and accepting eIDs issued in other Member States. Hence, this insufficient cross-border interoperability of national eIDs has prevented, until now, citizens and businesses from benefitting fully from the digital single market.

The Electronic Identification and Trust Services (eIDAS) Regulation[1] provides a solution to these issues by ensuring the cross-border mutual recognition of eIDs. Technical specifications and reference implementations of the interoperability nodes for the eID mechanisms were published

The associate editor coordinating the review of this manuscript and approving it for publication was Ranjan Bose.

[1] eIDAS Regulation: https://ec.europa.eu/digital-single-market/en/trust-services-and-eid

as open source on 26 November 2015 for the technological infrastructure under the Connecting Europe Facility (CEF) program.[2]

The ultimate goal of this initiative is to offer the possibility to EU citizens to use their national eID in other EU countries when accessing public and private services online. As stated in the eIDAS regulation, the mandatory mutual recognition of electronic identities in the whole EU is applied from September 2018.

In terms of privacy, eIDAS complies with the OECD (Organization for Economic Cooperation and Development)[3] recommendations, which defines a framework to improve privacy protection management and interoperability between OECD members [1]. Specifically, eIDAS regulation follows the OECD privacy principles of user data collection limitation and data quality, ensuring user data sovereignty.

In parallel, FIWARE Community[4] has emerged as an independent open community whose members are committed to materialise the FIWARE mission, that is: *to build an open sustainable ecosystem around public, royalty-free and implementation-driven software platform standards that will ease the development of new Smart Applications in multiple sectors*.

To achieve this mission, the FIWARE platform provides a rather simple yet powerful set of APIs (Application Programming Interfaces) that ease the development of Smart Applications in multiple vertical sectors and oriented to the Future Internet. The specifications of these APIs are public and royalty-free. Besides, an open source reference implementation of each of the FIWARE components (named Generic Enablers, GEs) is publicly available so that multiple FIWARE providers can emerge faster in the market with a low-cost proposition.

In this scope, FIWARE Security Framework [2], [3] offers a set of components that provide Identity and Access Control to the services and applications ecosystems developed and designed using the FIWARE APIs and components. Thus, every service developed in FIWARE platform, offers to its users the possibility to authenticate and secure their APIs exploiting the FIWARE Identity component.

Specifically, FIWARE Security framework offers an OAuth 2.0-based mechanism for external applications registration. On the other hand, to manage access control FIWARE Security GEs are compatible with a basic Role-based access control (RBAC) internal mechanism and with an advanced Attribute-Based Access Control (ABAC) mechanism based on XACML 3.0 [4].

OAuth 2.0 is the most extended standard for providing third party delegated authentication in the applications and services in the Internet [5]–[7]. Good examples of identity providers that supports this standard are Facebook, Twitter or Google.

In this paper we propose a model that enables the connection of OAuth 2.0-based services with the SAML 2.0-based eID authentication provided by eIDAS reference. Thanks to this model, services already connected with an OAuth 2.0 identity provider can be automatically connected with eIDAS nodes for providing eID authentication to European citizens.

For illustrating the implementation of the model we have integrated it with FIWARE Security GEs. Thus, these software components are connected with the CEF eID building Block to allow CEF eID transnational authentication of EU citizens by means of their national eID in FIWARE-based OAuth 2.0 authentication domains.

After the integration of FIWARE Identity Management and Access Control GEs with the CEF eID Building Block, every FIWARE ecosystem service can be securely accessed by Member States citizens. Furthermore, as stated before, FIWARE provides a Security Framework to enable service and applications development with OAuth 2.0-based authentication. Thus, every new service designed and deployed according FIWARE security basis, would be also accessible by European citizens using their eID.

For validating the proposed model we have deployed an instance of the FIWARE Identity Manager connected to the Spanish eIDAS node. Then, we have registered two services and extended their functionalities for enriching the user experience thanks to eID authentication. The first one is a private service for providing multi-user video conferencing rooms. The second one is a public smart city service that offers citizens several facilities exploiting the information provided by sensors deployed in Santander city. We have evaluated the integration of both services with the eID authentication with real users from seven different Member States.

The document is structured as follows. In Section II we provide an overview of available related works in the state of the art. In Section III we describe the necessary FIWARE and eIDAS background to understand the rest of the paper. Then, Section IV presents the proposed model and the specific implementation we have done in FIWARE. Section V introduces the two use cases for validating the solution, their deployments and the received input from users. Finally, Section VI concludes the work and opens future lines of research.

II. RELATED WORK

Over past decades the European Union has realized the importance of implementing a global infrastructure for electronic identification of citizens in multiple sectors such as eHealth or eBanking. The regulation of eIDAS could be considered as the heir of another European approach called STORK (Secure idenTity acrOss boRders linKed). The STORK project deployed an interoperability platform to allow cross-border authentication to Member States citizens by using their eID. Works [8]–[10] show a description of the whole architecture of STORK and provide important insights into the main goals in concern of security and identity management when deploying this kind of architectures.

[2] eIDAS Specification: https://joinup.ec.europa.eu/document/eidas-technical-specifications-v10

[3] OECD: https://www.oecd.org

[4] FIWARE: The open source platform for our smart digital future, https://www.fiware.org/

Authors of [8] organize identity management models into four categories situating STORK model (and consequently eIDAS) as a distributed and federated SAML-based approach. On the other hand, they elaborate on how national regulations imposition for explicit consent of users over their data could compromise user experience. However, this kind of issues are related with specific legal regulations that cannot be solved from a technical point of view.

Besides, [11] identifies key privacy and security as the main challenges to be addressed by STORK providing a number of main decisions adopted by the project. Among others, it stresses the importance of user-centric identity management in terms of user data control and the definition of authorization policies to protect the confidentiality of users. eIDAS was conceived as an evolution of STORK that aims to solve its main issues.

For supporting the migration to eIDAS, old STORK nodes could be connected to the new eIDAS ones in order to allow cross-border authentication between member states that only support one of the two standards. However, as STORK does not support ciphered connections between peers, it is necessary to encrypt the authentication requests to ensure the confidentiality of user data in the new eIDAS model [12].

Moreover, when talking about interoperability between identity management models, it is crucial to non-compromising the privacy-preserving. Authors of [13] propose a complex model that enables federation between identity providers avoiding the definition of a common set of attributes. On the other hand, some studies [14], [15] have stressed the importance of privacy and interoperability in the Cloud Computing context. Non-transparency due to the location of sensitive data storage makes it difficult to comply with the privacy laws of each country and to exchange these data with organizations and services.

Regarding the connection of service providers to the eIDAS nodes, each Member State has to decide the protocol to be used as the connector module is specific of each implementation. As explained by authors of [16] and [17], SAML 2.0 is the most extended protocol among Member States. However, as they also explain, this standard presents several limitations and hinders the integration of service providers.

In this scope, the Innovation and Networks Executive Agency (INEA) of the European Commission through its CEF programme is promoting initiatives that facilitate the integration of public and private service providers in the eIDAS infrastructure. Several works have analyzed the issue and proposed solutions to simplify the integration for application developers.

Authors of [18] adopt a context and process analysis framework to address the question of whether the eIDAS complements or challenges the national e-government initiatives, specifically the Estonia's e-residency project. They conclude that the Member States have to contribute to the fast implementation of the eIDAS. In their opinion this will be the most effective measure for achieving cross-border use of e-services.

On the other hand, authors of [19] discuss whether and how two approaches can be combined in order to provide services for electronic identification and authentication of entities. They highlight the creation, verification, validation and preservation of electronic signatures as well as the registered delivery of documents in an efficient manner using cloud computing techniques.

Finally authors of [20] point out that the inclusion of *something-you-have* authentication factor such as the mobile devices must be also taken into account to improve the integration of services. They also make an analysis on how the technologies and standards used for connecting services can definitely decide the future of the eIDAS vision.

In this sense, nowadays the *de facto* standard for third party delegated authentication in Internet is OAuth 2.0 [21]–[24], used by well-known identity providers like Google [23] or Twitter [25]. OAuth 2.0 enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service, or by allowing the third-party application to obtain access on its own behalf.

Comparing OAuth 2.0 to SAML 2.0, [26] enumerates the differences between the two protocols and explains possible vulnerabilities they have. It points out that OAuth 2.0 is lighter and more scalable than SAML and that it could be easier integrated in services and platforms in the Internet. Moreover, they explain how from architectural design and security perspective it fits better with the most relevant Federated Identity Management (FIdM) approaches.

FIdM has a lot of challenges that have to be addressed [27] but it definitively improves the user experience and usability of services, specially from the business perspective [28].

Regarding the possible security vulnerabilities, some works have proposed fixes to security [29]–[31] or performance [6] issues found in the OAuth 2.0 protocol. However, some of them are related with a non fully implementation of the standard [32], [33].

We can conclude that OAuth 2.0 protocol is definitively a good solution for providing delegated authentication to third party applications and services in the Internet. In our proposal, we present a model to offer a single entry point for registering services based on OAuth 2.0, making the details about the eIDAS nodes specific protocols transparent for the service providers.

III. FIWARE IDENTITY AND eIDAS BASIS

The CEF eID building block is a set of services (including software, documentation, training and support) provided by the European Commission and endorsed by the Member States, which helps public administrations and private service providers to extend the use of their online services to citizens from other European countries. This is accomplished through

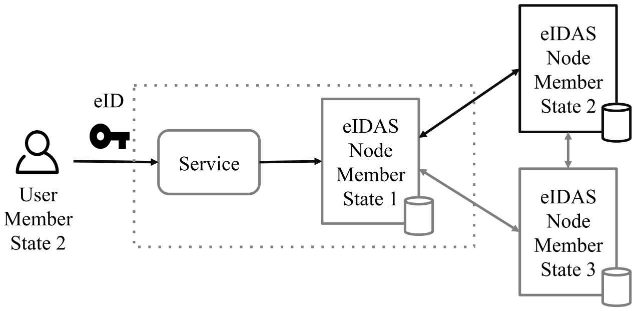


FIGURE 1. eIDAS basic architecture.

the mutual recognition of national eID schemes (including smartcards, mobile and digital certificates), allowing citizens of one European country to use their national eIDs to securely access online services provided in other European countries. The mutual recognition of eID schemes across Europe is mandated by the eIDAS Regulation.

In turn, the eIDAS Regulation states that all online public services requiring electronic identification assurance corresponding to a level of 'substantial' or 'high' must be able to accept the notified eID schemes of other EU countries. Public administrations offering online services that match these requirements are therefore obliged to comply. On the other hand, the specification is based on SAML 2.0 standard, so the integration is also open to private services that desire to offer their users the possibility of logging in by their eID. This could add an extra security level in some services that need to know exactly the identity of their users.

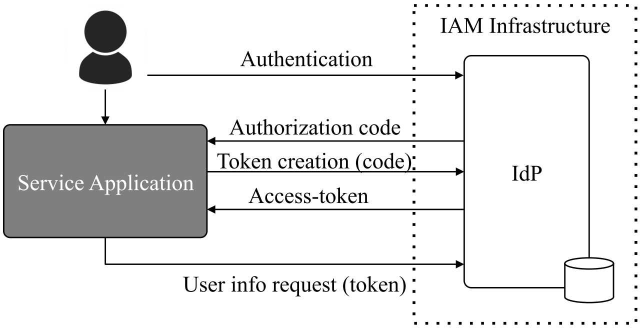


FIGURE 2. FIWARE IAM Architecture.

From a technical point of view, and supported by the deployed eIDAS Network, services deployed in a specific Member State can log in users of other Member States by means of their eID. As illustrated in Figure 1, a Service deployed in Member State 1 is connected to the eIDAS node of the same Member State. When a User of a different Member State tries to authenticate at that Service, the eIDAS node delegates the authentication request to the eIDAS node of the user's Member State. The protocol used by eIDAS reference for delegating the authentication is SAML 2.0 [34]. After a successful authentication, the Service receives a SAML Response containing the attributes of the user.

The objective of this work is to provide FIWARE Generic Enablers users the possibility of authenticating in the services provided by them using their national electronic identification. Currently, the FIWARE Identity Framework [3], [35] provides authentication to users by means of a username/password mechanism based on OAuth 2.0 standard [36].

Thanks to OAuth 2.0, users registered in a unique Identity Provider (IdP) can authenticate in external applications delegating the authorization to the IdP. Figure 2 shows how an external application can create a token (OAuth 2.0 *access token*) that represents the user in terms of authorization.

For creating this token any of the grants defined by the protocol can be used. The one that is relevant for eIDAS integration is the Authorization Code Grant. Using this grant and in authentication time, users are redirected to the IdP user interface where they securely enter their credentials. Once correctly authenticated, the IdP redirects the users to the application including an authorization code that they use to create the token. Once the token is created, it represents each user in terms of authorization and the application can access the public user information stored in the Identity Provider.

Furthermore, using the access token, the service can access other services and GEs ensuring that the resources exposed by them are accessed in a secure way. For achieving this, every request sent to a specific service is intercepted by a Policy Enforcement Point (PEP) that checks the user grants to access the resource. Figure 3 shows the rest of components available in the Identity and Access Control Management (IAM) infrastructure.

The Policy Administration Point (PAP) and the Policy Decision Point (PDP), together with the set of PEPs included in each Service, compose the widely-known Access Control architecture [37]. The PAP stores the defined access control policies in the Policies DB, where PDP checks them at decision time.

Figure 4 shows the flow for creating a token and using it to access a protected resource in a service. Once a user is registered in the IdP, it can create an OAuth 2.0 access token for accessing data in the Service backend. In OAuth 2.0 terminology that means creating a token in the scope of a consumer. This token represents the user in the system and has to be included in every request sent to the backend. As outlined before, these requests are intercepted by the PEP, that extracts the user's access token and validates it with the IdP. This validation can be performed using three levels of security:

1) Authentication: Using this level of security, the PEP just checks if the user has been correctly authenticated against the IdP. Thus, at this level, every user with an active account would be able to access the protected data. The check is performed by sending a validation request to the IdP.
2) Basic authorization: In this case, the PEP also checks if the user has the required permissions to perform the corresponding action (defined by an HTTP verb) in the corresponding resource (defined by an HTTP path). After the first check with the IdP, the PEP obtains the

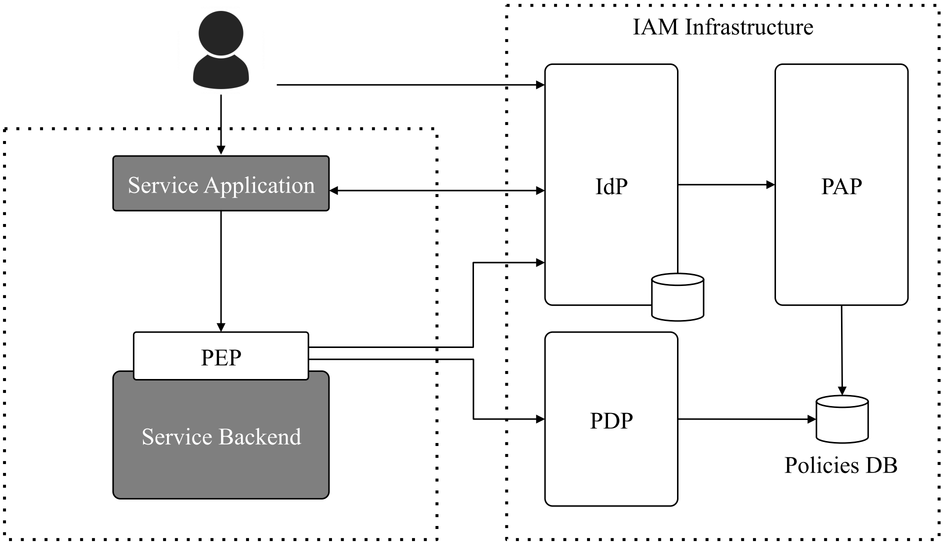


FIGURE 3. FIWARE IAM Architecture for securing backends.

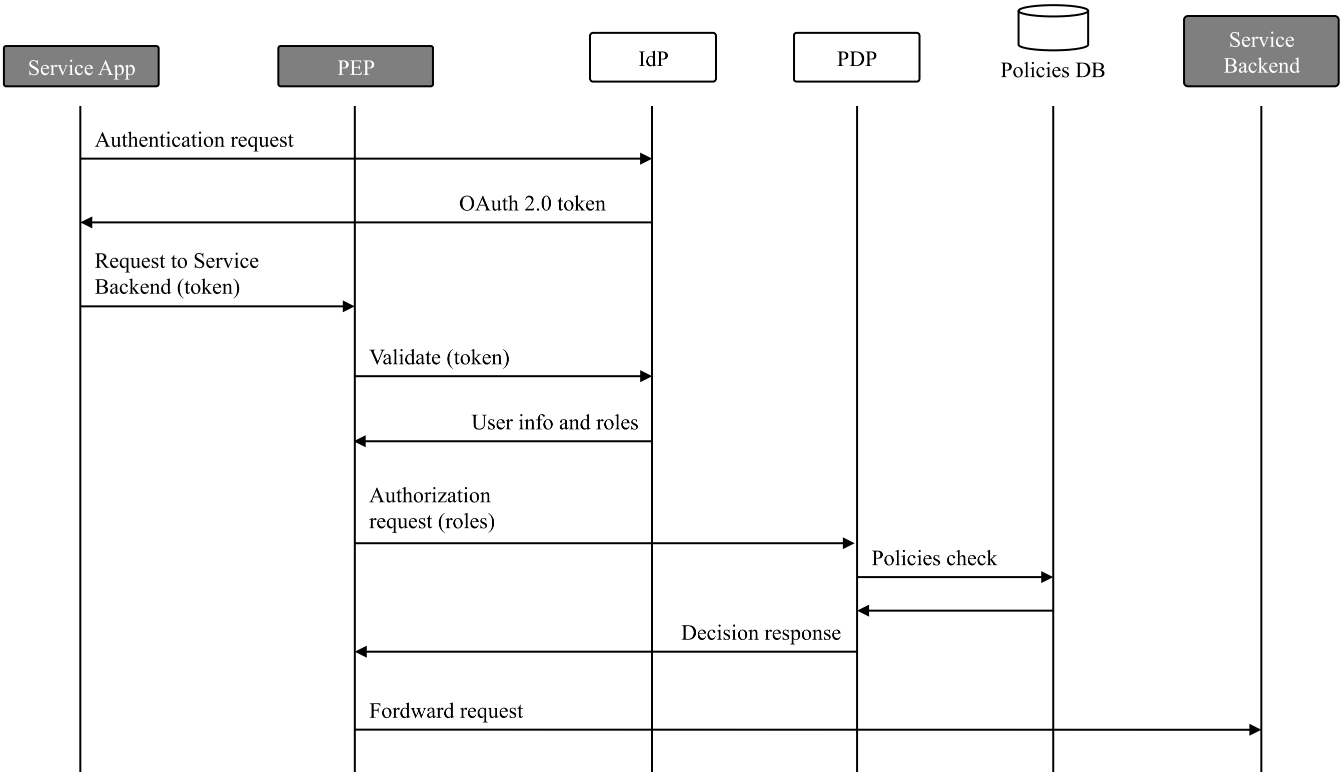


FIGURE 4. FIWARE Authorization flow.

roles (being a role a set of permissions) the user has been assigned in the scope of the Service where the token was created. Once roles have been retrieved, the authorization check is sent to the PDP. PDP fetches the policies associated with the user's roles from the Policies DB and decides whether or not access should be granted based on them.

3) Advanced authorization: This is the most complex, powerful case, because the authorization check is not only based on the HTTP verb and path, but also on other more advanced, customizable parameters, such as the request body or headers. To perform the check, a custom XACML policy request is sent to the PDP.

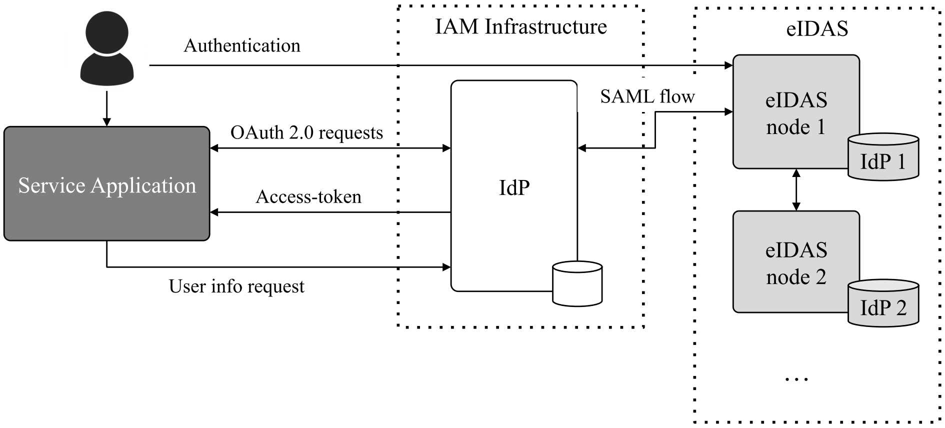


FIGURE 5. FIWARE IAM Architecture with connection to eIDAS.

IV. FIWARE IDENTITY FRAMEWORK WITH SUPPORT TO eIDAS

Being explained how eIDAS reference and FIWARE Identity work, in this Section we propose a solution for integrating both technologies and providing a way of authenticating European citizens in OAuth 2.0 services by means of their eID. Our proposal is designed for being compatible with any OAuth 2.0 server. In the following, we first explain how the generic model works and then we describe its integration in the specific case of FIWARE Identity Manager.

A. OAuth 2.0 - eIDAS MODEL

For supporting OAuth 2.0 authentication using the eID of citizens, we propose a gateway between OAuth 2.0 and eIDAS SAML 2.0. The main requirement of this gateway is to be able to include eIDAS users in an OAuth 2.0 paradigm as the one explained above. Coming back to Figure 3, when the Service Application authenticates a user, it would be not required that the user is already registered in the IdP. Users with a valid eID could directly log in the service and obtain an OAuth 2.0 access token that represents them in terms of authorization.

Figure 5 shows the model we propose for supporting this requirement. The Identity Provider is connected to the eIDAS node of the country where it is deployed. Therefore, it can send SAML 2.0 Authentication Requests to the eIDAS infrastructure on behalf of the service. Leveraging the eIDAS nodes network, if the user that is trying to authenticate belongs to a different country, the eIDAS nodes will interchange the needed SAML requests between them.

For being able to authenticate a user using SAML, services have to expose a SAML metadata file with some information such as the public signing certificates that the eIDAS node will validate before the authentication. In our proposal, when a service is registered as an OAuth 2.0 consumer in the Identity Provider, it creates the metadata file for the service making this process transparent for developers. Thus, when a developer registers a new OAuth 2.0 consumer in the Identity Provider, it generates the OAuth 2.0 client and secret identifiers for creating access tokens and the SAML metadata file. This file can be provided to the service or directly served by the Identity Provider.

Figure 6 shows the requests flow interchanged between the entities of the model once the service has been registered in the IdP and when a citizen authenticates there using the eID.

1) The citizens try to log in the application and the application redirects them to the Identity Provider using OAuth 2.0 authentication flow. In this scenario the OAuth 2.0 Authorization Code Grant is used because is the one that uses a client side user interface for authentication.
2) The IdP authentication panel provides to the user the possibility of logging in using the traditional user/password credentials but also a new possibility of using the eID. If the second option is selected the IdP generates a SAML *AuthnRequest* and sends it to the eIDAS node on behalf of the service.
3) When the eIDAS node receives the SAML request, it checks if the user belongs to its country. If the answer is yes, then it authenticates the user in the local national Identity Provider where the citizen is registered. If the citizen belongs to a different country, it delegates the SAML request to the corresponding eIDAS node.
4) In both cases the final result is that the eIDAS node returns a SAML Response to the Identity Provider with the citizen's profile. Of course, every sensitive data interchanged between the eIDAS nodes and the IdPs is encrypted and signed. The components use SAML metadata files to retrieve the public certificates of each node for managing encryption.
5) When the IdP receives and decrypts the SAML Response, it extracts the user profile with the set of attributes available in the citizen's eID. Using this profile the IdP checks if the user is already registered or if a new account has to be created. Then, it performs a mapping between the attributes available in the citizen's eIDAS profile and the ones available in the IdP.

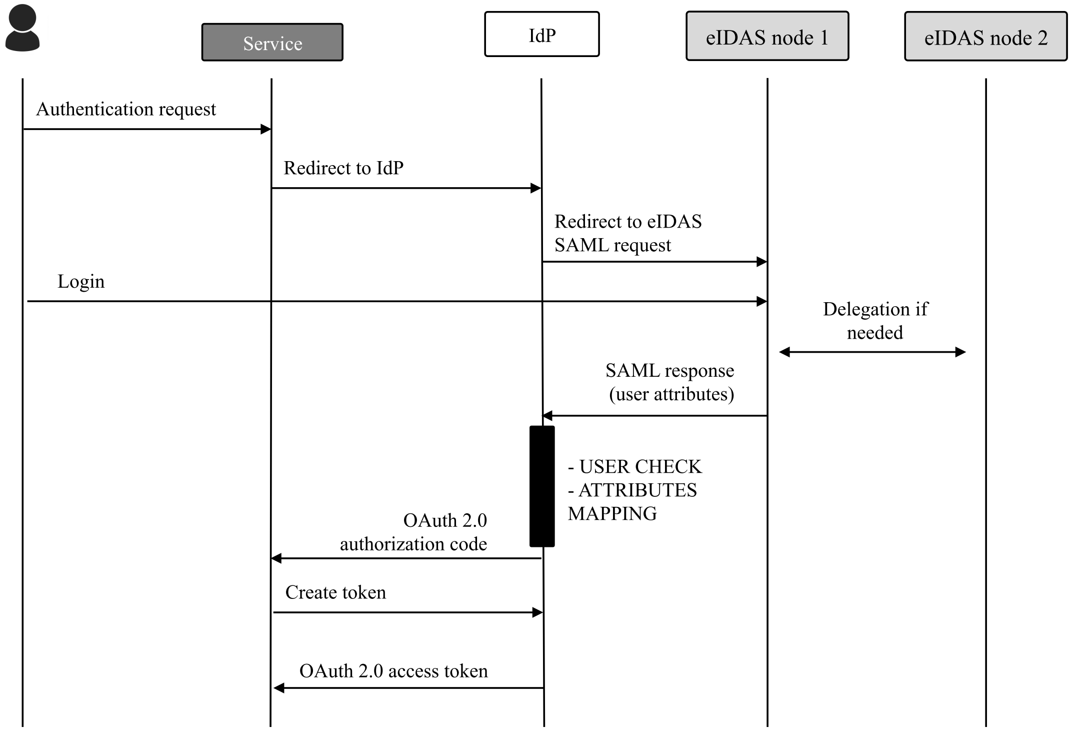


FIGURE 6. **FIWARE Authorization flow with connection to eIDAS.**

This process is explained with detail below. Finally, the IdP generates an OAuth 2.0 authorization code in the same way it does for a regular registered user.

6) With the authorization code the service can continue the OAuth 2.0 flow to create the access token that represents the user and that can be used for getting the public information or authorizing it for accessing any other service protected by a PEP.

With regard to attributes mapping, it is important to take into account that the eIDAS profile could not provide a unique identifier for citizens. For some Member States, the eIDAS attribute *PersonIdentifier* can unequivocally identify a citizen. However, this is not valid for all of them. For instance, for German citizens, the provided *PersonIdentifier* attribute changes every time the citizen receives a new eID token (i.e. smart card). Likewise, Italian citizens can have multiple *PersonIdentifier*, as the Italian national eID system supports multiple IdPs, each providing a different *PersonIdentifier* for the same citizen. Therefore, citizens have to be identified in the IdP database using the same unique identifier than regular users (typically a *uuid*, username, email or phone number). Furthermore, the solution has to support the possibility of associating several eIDAS *PersonIdentifier* to the same user account. In our model, as we will justify later, this unique identifier has to be remembered by users. Thus, a username, an email or a phone number are better options than *uuid's*. Email addresses and phone numbers allow also the possibility of getting confirmations from users, for instance for creating an account or restoring passwords. For facilitating the reading, in the following paragraphs we will generalize this unique identifier as a *username*.

Taking this into account, when checking if the user that is trying to log in is already registered in the IdP (step 5), the following situations have to be considered.

- New user: when a new user tries to authenticate, the IdP has to create a new account in the database using a username. If the user consents the creation of the account, the IdP has to associate the eIDAS profile to the new account including the eIDAS *PersonIdentifier* attribute.
- User already registered in the IdP with username/password: when the user trying to authenticate has previously created an account in the IdP using username and password, the eIDAS profile has to be associated to this account. For achieving this, the user has to provide the username of the existing account. Then, the IdP is able to associate the *PersonIdentifier* and add the eIDAS profile of the user to the existing account.
- User already registered in the IdP with an eIDAS identifier: when a user trying to authenticate has previously created an account in the IdP using an eIDAS profile with the same *PersonIdentifier*, the IdP has to use the same account for authenticating the user.
- User already registered in the IdP with a different eIDAS identifier: when a user trying to authenticate has previously created an account in the IdP using an eIDAS profile with a different *PersonIdentifier*, the new *PersonIdentifier* has to be associated to this account. For achieving this, the user has to provide the username of

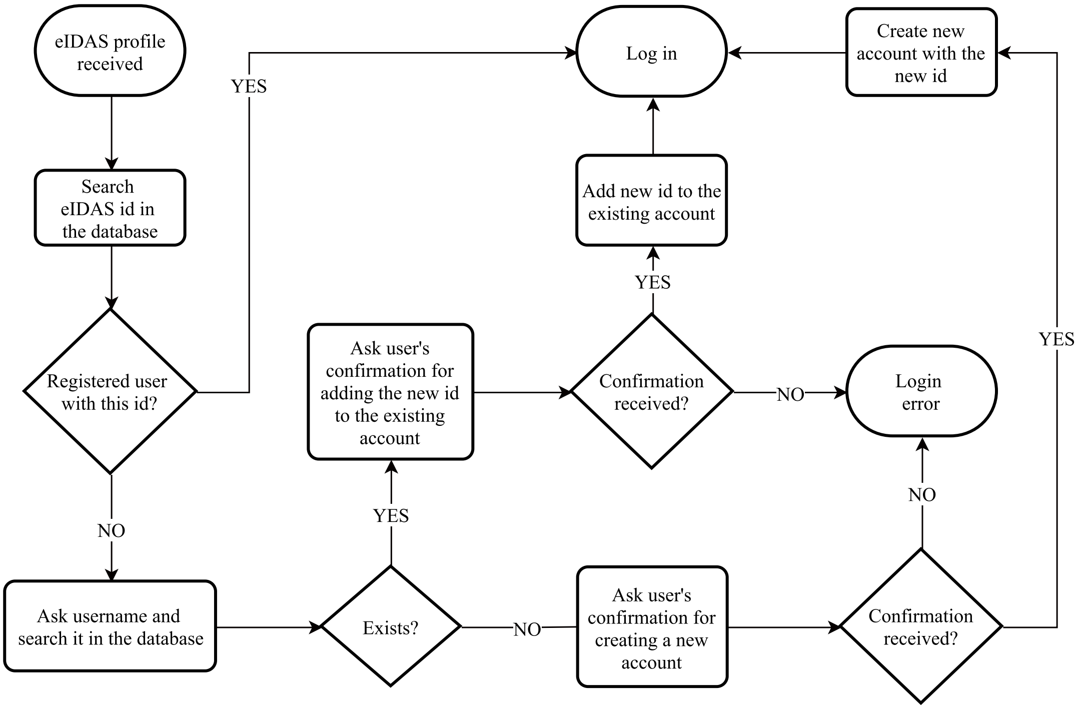

eIDAS profile received
Search eIDAS id in the database
Registered user with this id?
YES
NO
Log in
Create new account with the new id
Add new id to the existing account
YES
Ask user's confirmation for adding the new id to the existing account
Confirmation received?
NO
Login error
YES
YES
NO
Ask username and search it in the database
Exists?
NO
Ask user's confirmation for creating a new account
Confirmation received?

FIGURE 7. Workflow to map an eIDAS profile in the IdP.

TABLE 1. Mandatory attributes in eIDAS nodes.

PersonType	FriendlyName	NameUri
NaturalPerson	PersonIdentifier	http://eidas.europa.eu/attributes/naturalperson/PersonIdentifier
	CurrentFamilyName	http://eidas.europa.eu/attributes/naturalperson/CurrentFamilyName
	FirstName	http://eidas.europa.eu/attributes/naturalperson/FirstName
	DateOfBirth	http://eidas.europa.eu/attributes/naturalperson/DateOfBirth
LegalPerson	LegalPersonIdentifier	http://eidas.europa.eu/attributes/legalperson/LegalPersonIdentifier
	LegalName	http://eidas.europa.eu/attributes/legalperson/LegalName

the existing account. Then, the IdP is able to associate the new *PersonIdentifier* to the existing account. In this case, the user will have several *PersonIdentifier* associated with the account.

Figure 7 illustrates the process we propose for managing each of these situations. When receiving the eIDAS profile of the user that is trying to authenticate, the IdP searches the *PersonIdentifier* extracted from the eIDAS profile in the database. If a user with such identifier already exists, the authentication process is complete. If it does not exist, three possibilities may occur: 1) new user, 2) user already registered with username/password or 3) user already registered with a different eIDAS *PersonIdentifier*. For checking this, the IdP asks the user to provide a username and searches it in the users database table. If the user already exists, the IdP asks the user to confirm the association of the new eIDAS profile to the existing account (for username/password accounts the eIDAS profile is created and for accounts already associated to an eIDAS identifier the profile is updated with the new *PersonIdentifier*). Asking the user's confirmation for creating or linking the account avoids identity theft and could be performed by sending an email, a message, by signing a consent, etc. If the provided username does not exist, the IdP asks the user consent to create a new account and to associate the eIDAS profile using the same procedure.

When the IdP creates a new account for the user, it has to perform a mapping between the received attributes and the ones defined in the IdP user schema. eIDAS specification defines a set of attributes supported by the eIDAS nodes. However, not all of them are mandatory so depending on the specific eIDAS node they could be supported or not. Table 1 shows the mandatory attributes that every eIDAS node has to support. This means that the IdP has to perform a mapping at least between this set of attributes and the user schema of its database.

In our model we propose to include an extra JSON (JavaScript Object Notation) field in the user data schema for storing the complete eIDAS profile of the user apart from the directly mapped attributes. For instance, it is very common that a user has a field for the family name or the date of birth, but not for the legal identifier. In this case the family name and the date of birth retrieved from the eIDAS profile would be directly stored in the IdP user profile, while the rest of attributes would be stored in the extra field as a JSON object. An example of this attributes mapping is illustrated in the next subsection. The information about the eIDAS profile of the user can be included as part of the user information provided by the IdP when an application requests the user info using an OAuth 2.0 token.

As for data protection, it is important that the Identity Manager shows a consent informing users that their eID profile will be stored and managed by this new entity. Specific data protection regulations of each Member State would be applied for storing and using these data.

B. FIWARE IMPLEMENTATION

We have implemented the proposed model as an extension of FIWARE Keyrock Generic Enabler. Keyrock is the Identity Management GEri (Generic Enabler reference implementation) of FIWARE[5] and it brings support to secure and private OAuth 2.0-based authentication of users and devices, user profile management, privacy-preserving disposition of personal data, Single Sign-On (SSO) and Identity Federation across multiple administration domains.

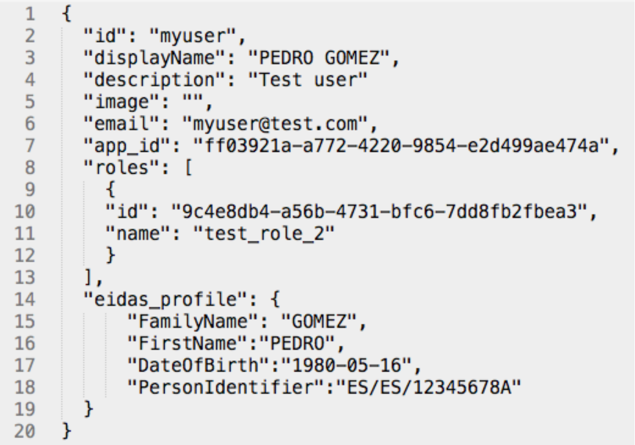
Keyrock has been developed by part of the authors of this paper using Node.js and Express and makes use of an SQL database for persistence. For implementing the OAuth 2.0 - eIDAS model we have extended an existing SAML 2.0 library[6] for managing the authentication flow between the IdP (Keyrock) and the eIDAS nodes. Moreover, we have extended the OAuth 2.0 module of the components for enabling the possibility of registering OAuth 2.0 consumers as Service Providers in the eIDAS nodes. Thus, when developers register a new consumer, they can choose the option of enabling the eIDAS connection. After introducing the service information in a web form, Keyrock generates and serves a metadata file that will be checked by the eIDAS node for getting the needed information (certificates, authentication callback URLs, etc).

TABLE 2. Relevant attributes in Keyrock's user profile.

Name	Description
id	UUID that uniquely identifies a user in Keyrock
displayName	Friendly Name that identifies a user in Keyrock
description	Text string which the user can use to include information about themselves
image	Path to the image of the user
email	Mail address of the user. It is unique in the database
password	Private key for authentication
enabled	Boolean that allows the user or not to perform requests to Keyrock
admin	Boolean that indicates if the user is an administrator of Keyrock
extra	Allows to store extra attributes of the user

Table 2 shows the relevant existing attributes in Keyrock's users schema. When Keyrock receives a SAML Response from the eIDAS node with the eID profile of a citizen it executes the process explained in the previous subsection to check the nature of the user. In Keyrock, the email of the citizens is used as unique identifier and unequivocally identifies them in the database. The email address is also used for notifications and password reset purposes. Therefore, when asking for the username in the specific step of Figure 7, citizens have to provide their email address to check if their account already exists in the database. To avoid identity theft, they receive a confirmation email that has to be acknowledged before continuing the process. In case the eIDAS profile of a citizen has to be created or updated in Keyrock, the following actions are performed:

1) Attribute *displayName* is created as a concatenation between the eIDAS attributes *FirstName* and *CurrentFamilyName*.
2) If available, the user photography is stored in *image* attribute.
3) The eIDAS attribute *PersonIdentifier* is stored in a new attribute named *eidas_id*. Citizens could have more than one *PersonIdentifier* associated to their accounts in case they are not unique in the specific Member State.
4) The whole eIDAS profile is stored in the *extra* field as a JSON object called *eidas_profile*



```
{
  "id": "myuser",
  "displayName": "PEDRO GOMEZ",
  "description": "Test user"
  "image": "",
  "email": "myuser@test.com",
  "app_id": "ff03921a-a772-4220-9854-e2d499ae474a",
  "roles": [
    {
      "id": "9c4e8db4-a56b-4731-bfc6-7dd8fb2fbea3",
      "name": "test_role_2"
    }
  ],
  "eidas_profile": {
      "FamilyName": "GOMEZ",
      "FirstName":"PEDRO",
      "DateOfBirth":"1980-05-16",
      "PersonIdentifier":"ES/ES/12345678A"
  }
}
```

FIGURE 8. User profile returned by Keyrock.

As explained above, when an application requests the user information using an OAuth 2.0 access token, the eIDAS profile of the user is included in the response. Figure 8 shows

[5] https://fiware-idm.readthedocs.io

[6] https://www.npmjs.com/package/saml2-js

and example of the JSON returned by Keyrock when an application validates a user access token.

In the example, Keyrock has mapped the name of the user and has included the received eIDAS profile in the *extra* field. On the other hand, Keyrock returns the roles that the user has been assigned in the scope of the application. These roles will be used for enforcing the actions the user can perform in the application.

V. USE CASES VALIDATION

We have deployed an instance of Keyrock Identity Manager and registered two services to offer a testing environment to a set of users from different European countries. These users have tested the services and provided feedback answering a survey. This section describes the details of the deployed services and the results obtained from the evaluation.

A. IDENTITY MANAGER DEPLOYMENT

An instance of Keyrock Identity Manager component has been deployed for performing the tests and for registering the use cases service providers. This instance includes the implementation of the OAuth 2.0 - eIDAS model proposed in this paper.

The instance is publicly available[7] and it has been deployed in a virtual machine with the following characteristics:

- Operating system: Ubuntu 16.04
- RAM: 4GB
- VCPUs: 2 VCPU
- Disk: 40GB

The instance is connected to the official Spanish eIDAS node.[8] This node is only accessible by authorized service providers so both use case pilots have been registered as applications in Keyrock and also in the eIDAS node. After registering the services, Keyrock serves the corresponding metadata files so the eIDAS node can validate them and get the public signing certificates.

B. MASHMETV VIDEOCONFERENCING SERVICE

The first service we have deployed is a private business video conference platform, mashme.io. It is used in many different verticals, like e-learning, consulting, e-health, etc. In all those markets, users ask for an integrated experience with their existing platforms, thus Single Sign On and centralized identity is a must for the product.

Mashme.io is a Service-as-a-Service video collaboration platform which is delivered via a web browser and requires no plug-ins or application installation on the desktop of clients. The software is based in HTML5 and WebRTC (Web Real Time Communication) standards.

The aim of the product is to solve the problem of fragmented tools and ephemeral discussions by offering a complete real-time synchronized virtual classroom solution. It is a modular platform that can be adapted to different business needs.

1) INTEGRATION WITH FIWARE-eIDAS SOLUTION

Taking advantage of the eID-FIWARE IdM Security framework that offers an OAuth 2.0-based mechanism to external applications, the first step is to register the mashme.io web service[9] as an application at the eID-FIWARE IdM. For this purpose a web service URL is provided as well as a redirect callback URL.

Registering as an external application at eID-FIWARE IdM grants the mashme.io service OAuth 2.0 credentials that will identify the service through the Identity Manager. Thus, being connected with an OAuth 2.0 identity provider, thanks to the eID-FIWARE IdM Security framework, the mashme.io service will be automatically connected with the eIDAS nodes. This will provide mashme.io an eID authentication option to all our European users.

Due to the private nature of the service, it is necessary to have an active mashme.io account for using the service. Thus, in order to provide a testing environment on the mashme.io service, we have created a special organization called FIWARE, were all pilot users will belong to. This test environment is the same as the one in production but with the option to vinculate the mashme.io user account with the eID-FIWARE IdM.

Then, mashme.io FIWARE users will have a link on their mashme.io accounts to vinculate the account to eID-FIWARE IdM. When this process starts, mashme.io, as an external registered application using its OAuth 2.0 credentials, will start the authentication process redirecting the user to the FIWARE IdM. Then, through the gateway between OAuth 2.0 and eIDAS SAML 2.0 provided by the eID-FIWARE IdM, users will be able to log in using their eID. Once the flow between Keyrock and the eIDAS node is managed by the gateway the user identity profile is retrieved from the corresponding eIDAS node. Afterwards, the profile is used by Keyrock to create a local copy of the user and to generate an OAuth 2.0 authorization code.

With the authorization code the mashme.io service continues the OAuth 2.0 flow creating the access token that represents the user in the service and linking it to the mashme.io account. Accordingly, the mashme.io service is able to use the information retrieved from the eIDAS profile of the citizen to personalize the service.

2) eIDAS ATTRIBUTES USED

Mashme.io highlights the value of the data received from eIDAS and FIWARE IdM to improve, detail and deepen the use case. Below we show how to make use of the data provided in our platform.

- Platform Integration: The following data of the eID profile is stored in mashme.io: *PersonIdentifier*, *CurrentFamilyName*, *FirstName* and *DateOfBirth*
- User Profile: To enrich the user profile and provide a more tailored experience, mashme.io makes use of the following attributes: *PersonIdentifier*, *FirstName*

[7] Keyrock Identity Manager instance: https://idm-cef-fiware.dit.upm.es

[8] Spanish eIDAS node: https://se-eidas.redsara.es/EidasNode

[9] Mashme.io service: https://app.mashme.io/login-fiware/login.html

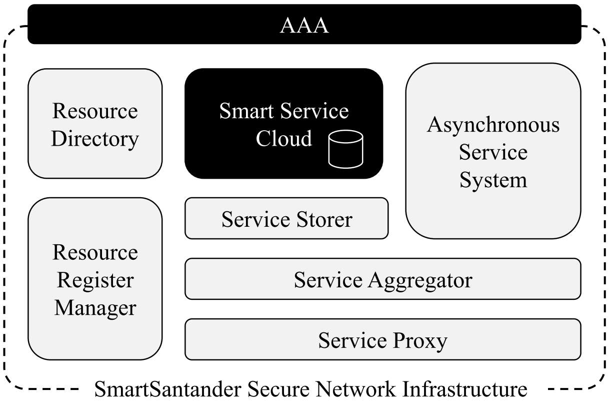

AAA
Resource Directory
Smart Service Cloud
Asynchronous Service System
Service Storer
Resource Register Manager
Service Aggregator
Service Proxy
SmartSantander Secure Network Infrastructure

FIGURE 9. SmartSantander facility architecture.

and *CurrentFamilyName* (to create a mashme.io name attribute). The attributes extracted from the Keyrock profile are the email, the image (we will need to adapt the image data to our avatar format) and the *enabled* attribute (based on the enabled attribute we are be able to activate the account at mashme.io or disable it with our type flag).

- Customized Platform: Parsing eIDAS user profile *PlaceOfBirth* and/or *CurrentAddress* attributes, we could get the information regarding users native language and set up the platform with its language if available.
- Billing Integration: In order to facilitate access and commitment of our customers with our platform, the following attributes provided by eIDAS would be used to register a mashme.io billing account profile speeding up the set up and integration process: *LegalName*, *LegalPersonAddress*, *VATRegistrationNumber* and *TaxReference*.

C. SMARTSANTANDER SMART CITY

The second service we have deployed for validating our proposal is based on the already existing Smart City infrastructure called SmartSantander. This facility is based on a real IoT deployment in an urban setting. The core of the facility is located in the city of Santander (Spain) and surroundings, embracing IoT deployments in different key areas of the city infrastructure, ranging from public transport, key logistics facilities, public places and buildings, workplaces and residential areas, thus creating the basis for the development of a smart city [38]. This deployment exhibits the diversity, dynamics and scale that are essential for the creation of digital solutions that address urban challenges.

The deployed facility has a dual purpose. On the one hand, it enables real-world experimentation on IoT related technologies (protocols, middleware, applications, etc.). On the other hand, it supports the provision of smart city services aimed at enhancing the quality of life in the city of Santander. In this sense, as described in [39], a large number of added-value services have been developed on top of the facility.

In Figure 9 we present a general representation of the main components of SmartSantander facility. First, a set of components enables the integration of IoT services and devices (Service Proxy and Aggregator). Then, the data provided by the integrated services is stored in the Smart Service Cloud, which is the actual data repository. The stored data can be afterwards consumed, leveraging different services, through the security layer. As can be observed, the SmartSantander platform exposes an Authentication, Authorization and Accounting (AAA) layer that permits controlling the access to resources and ensuring secure interactions with users and services. This layer belongs to the so-called SmartSantander Secure Network Infrastructure, which implements various network security mechanisms to prevent unauthorized access and attacks. Being part of a larger security framework, the AAA layer is tightly coupled with the components that provide support to SmartSantander services. Therefore, in order to validate the integration with FIWARE-eIDAS solution we have deployed a parallel testing infrastructure that directly accesses to the core of SmartSantander components.

1) INTEGRATION WITH FIWARE-eIDAS SOLUTION

As explained above, we have deployed a testing infrastructure that gets direct access to the Smart Service Cloud component. To validate the integration we have developed an IoT data browsing web application, which is developed on top of the FIWARE PEP Proxy in order to provide secure access to the stored data.

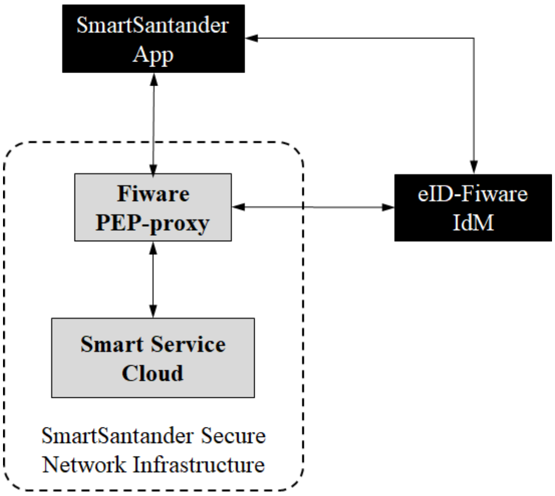


FIGURE 10. SmartSantander Integration platform.

Figure 10 shows the software platform developed to carry out the integration of the FIWARE-eIDAS solution into SmartSantander framework. The SmartSantander web application[10] interacts with the deployed instance of Keyrock to obtain an access token, which is afterwards used by FIWARE PEP-Proxy as a key to open the access to Smart Service Cloud.

[10] SmartSantander web application: https://eid-fiware.smartsantander.eu

The web application is registered in eID-FIWARE IdM in order to be recognized as an OAuth 2.0 client. First, it must be specified the URL where the application will be executed and the callback URL, which is the URL where the application will be redirected to when the OAuth 2.0 process ends. As a result, eID-FIWARE IdM provides the OAuth 2.0 credentials of the application (client ID and client secret), which will be used when the application requests access to SmartSantander services.

Then, a FIWARE PEP Proxy is registered in order to enable authentication and authorization via OAuth 2.0 to the web application. The credentials generated during this step allow eID-FIWARE IdM to recognize the FIWARE PEP-Proxy as an authorized instance to validate the access token used by the web application. The access token obtained during the user authentication is included in each request of the web application. In this sense, when the FIWARE PEP-Proxy receives the request, it validates the access token with the eID-FIWARE IdM. If the token is valid, the PEP-Proxy forwards the request to Smart Service Cloud to retrieve the dataset of the corresponding service.

2) SMARTSANTANDER WEB APPLICATION

It is the visible part of the integration of SmartSantander platform with FIWARE-eIDAS solution. This web application has been implemented in two parts: frontend and backend. The frontend is the part of the application with which the user interacts to access the services of SmartSantander using a web browser. It has been developed with React JS and other additional libraries that implement its functionalities, for instance axios, leaflet, reactstrap, etc.

The backend implements the API that interacts with both the web page and the security components, FIWARE PEP-Proxy and eID-FIWARE IdM. The frontend sends the requests to the backend and the latter, previously processed, redirects them to the corresponding FIWARE components. The backend has been implemented with Node JS and Express JS, and it uses MongoDB to store the necessary information to generate the usage statistics of SmartSantander services. It is worth highlighting that the API does not allow to store sensitive information of the user, and also implements routines that prevent from providing such sensitive information to the frontend, for example eID number.

3) eIDAS ATTRIBUTES USED

When users authenticate to FIWARE IdM using their national eID, an IdM profile is created with the user's personal information provided by the eIDAS node corresponding to the eID of the user. These are the eIDAS attributes, which can be used by applications and services to customize their functionalities based on the user profile, obtain more reliable usage statistics, etc.

One of the main objectives of SmartSantander is to boost the use of the platform among the scientific community, end users and service providers in order to reduce technical and societal barriers that prevent the IoT concept to become an everyday reality. Taking into account this objective, services and applications were developed based on more relevant use cases of the city in order to achieve the greatest scope and impact on citizenship. For this reason, the services and applications were not designed with characteristics that enable its use over specific groups of users.

Although SmartSantander services are not susceptible to be customized based on the attributes available in eIDAS user profile, the SmartSantander web application includes three functionalities that demonstrate its integration with eIDAS:

- Welcome message: Once the user has been authenticated, the eIDAS attribute *FirstName* is used to display a welcome message with the real name of the user. In addition, based on the eIDAS attribute *PlaceOfBirth* a flag icon corresponding to the user's country is displayed.
- User activities: The eIDAS attribute *PersonIdentifier* is used to track the user activity inside the platform. As this attribute could not unequivocally identify users, the unique identifier provided by Keyrock is used together with it.
- Usage statistics: The eIDAS attributes *DateOfBirth* and *PlaceOfBirth* is used to collect statistical data about the use of the web application. This information could be used to build strategies to foster the use of the services.

TABLE 3. Nationality of the users.

Nationality	Number of users
Spain	6
Portugal	1
Italy	1
Germany	1
Czech Republic	1
Slovenia	1
Austria	1
Test user	7
Total	**19**

D. RESULTS

For validating the deployed pilots we have found a set of users representing seven European countries. The requirement for using the services and evaluating the proposal is to own a valid and up-to-date digital authentication mechanism of the specific country. Depending on the country the notified authentication schema can be an electronic card, a digital certificate or a mobile-based authentication method. Table 3 shows the distribution of citizens between the countries. Apart from the citizens from these seven countries, we have provided a testing certificate to seven additional users that have tested the pilots too.

Before testing the pilots, users had to answer a first question about the convenience of having a common identification point for European citizens. Then they had instructions to access each service, log in using their eID and explore how their digital profile is used for the enrichment of the services.

TABLE 4. Survey answers.

Question	Yes	No	I'm not sure
1. I think that creating a common identification and access point for accessing European services and applications with the eID of citizens will improve the usability of those services	94.7 %	5.3 %	0 %
2. I have used SmartSantander service and I think that the possibility of logging in there with my citizen eID facilitates the access and improves the user experience	84.2 %	5.3 %	10.5 %
3. I have used mashme.io service and I think that the possibility of logging in there with my citizen eID facilitates the access and improves the user experience	57.1 %	14.3 %	28.6 %
4. After testing this pilot, I would like the inclusion of this initiative in other public and private services in Europe	89.5 %	0 %	10.5 %

Spanish users directly authenticate in the Spanish eIDAS node after the redirection from Keyrock. However, other Member State citizens are redirected to the specific eIDAS node of their countries by the Spanish one validating the cross-border authentication mechanism provided by eIDAS. After the experience, they had to answer three additional questions asking for their opinion about each service and the generic integration of the proposed solution in other services.

Table 4 shows the questions of the survey and the answers provided by the users. As explained before, the first question was answered before testing the services and the answers are clearly positive. Regarding Smartsantander service, we observe that users have mostly noticed that the use of our solution for integrating eIDAS authentication in this application improves the experience and facilitates the access. However, in mashme.io service the results are not so good. This is probably due to the fact that users had to request an account before start using the service. We conclude that it is not enough with providing to developers the mechanisms to integrate eIDAS authentication in their services, it is also very important the way in which these facilities are finally integrated in each specific service.

Nevertheless, the global opinion about the initiative is very positive and almost 90 % of the users would like the inclusion of this solution in other services and applications in Europe. It is also relevant that the small part of users that refuse the idea at the beginning, have now doubts about the convenience of its massive implementation.

Finally and regarding the amount of users that have validated the pilots, the hard requirements for using the eIDAS enabled services nowadays are the main cause of such number. As stated before, for using the services citizens have to own a valid and up-to-date digital authentication mechanism of their country. On the other hand, the eIDAS network established between Spanish node (where we have connected the services) and the citizens' Member State has to be established, up and running. However, the most important point to be addressed is to validate that services can be connected to the eIDAS infrastructure in an easy and transparent way for developers. In that sense, the number of countries tested is rather sufficient and demonstrates that the eIDAS initiative is being taken seriously by Member States and that citizens are positive with the idea of having a common identification schema for accessing public and private services in Europe.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we have proposed a model that facilitates the connection of private and public service providers to the European eIDAS infrastructure. The model allows developers already using an OAuth 2.0-based delegated authentication mechanism to offer to their users the possibility of logging in using their national eID. As explained above, OAuth 2.0 is the *de facto* mechanism to delegate authorization in third party applications and it is currently used by identity managers of platforms such us Facebook, Twitter or Slideshare.

Consequently, our proposal enhances interoperability of the eIDAS infrastructure by abstracting the connection of service providers through a component acting as a gateway to OAuth 2.0. Thanks to the application-scoped property of the model, citizens' privacy is preserved. Furthermore, users must give explicit consent to share their data when authenticating to each specific service following OAuth 2.0 standard.

The model has been implemented in FIWARE Identity Manager but it can be also used for enabling the connection to the eIDAS infrastructure in any other OAuth 2.0 identity server. This is a very interesting advantage in a moment in which the European Commission is putting a lot of efforts on exploiting the use of the eIDAS infrastructure in public and private services among Europe.

For validating our proposal we have deployed an instance of the FIWARE implementation of our model and we have connected two service providers that are publicly available. MashmeTV provides a videoconferencing tool for e-learning, consulting and e-health based on WebRTC standard. Thanks to the proposed integration users can link their Mashme accounts to their national identity for personalizing the experience and providing personal and billing information.

On the other hand, Smart Sandander provides access to the smart city IoT deployment at Santander city in Spain. This deployment enables services for tourism, traffic control or parking for citizens. After the integration with eIDAS, European users can use their eID for accessing these services and getting a personalized experience.

Nineteen users from seven different Member States have tested the deployed services providing us with their impressions about the experience being the received feedback very positive. After testing the pilots, almost the 90 % of the citizens think that the inclusion of this initiative in other public and private services in Europe will definitely facilitate the access and improve the user experience.

The proposed model could be improved by supporting compatibility with other well-known standards such as OpenID Connect. It is also interesting to study the integration with clients that support JSON Web Tokens or other authentication standards. This could enhance the proposal in terms of interoperability. The implementation could also be evolved in the future by testing it with eIDAS nodes of other Member States. This is important because the connector part of the eIDAS nodes is specific for each Member State. Thus, small adaptation could be needed in each implementation. Having a single identity manager reference that supports several connection standards and that is compatible with every eIDAS node implementation is a very interesting future work.

Finally, the implementation of the OAuth 2.0 - eIDAS model could be extended to support the connection to external attribute providers. This would add certified sources of users' personal data such as universities, medical systems or organizations for the social inclusion of people with disabilities. So that, services could use these data to provide services adapted to functional capabilities of users. This could also ease administrative tasks in public institutions of Member States.

REFERENCES

[1] (2013). *The OECD Privacy Framework*. Accessed: Nov. 6, 2019. [Online]. Available: http://www.oecd.org/sti/ieconomy/oecd_privacy_framework.pdf Last

[2] A. Alonso, A. Pozo, J. M. Cantera, F. de la Vega, and J. J. Hierro, "Industrial data space architecture implementation using FIWARE," *Sensors*, vol. 18, no. 7, p. 2226, 2018. [Online]. Available: http://www.mdpi.com/1424-8220/18/7/2226

[3] F. Fernández, A. Alonso, L. Marco, and J. Salvachúa, "A model to enable application-scoped access control as a service for IoT using OAuth 2.0," in *Proc. 20th Conf. Innov. Clouds, Internet Netw. (ICIN)*, Mar. 2017, pp. 322–324.

[4] (2013). *Extensible Access Control Markup Language (XACML) Version 3.0*. Accessed: Feb. 20, 2019. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html

[5] M. Darwish and A. Ouda, "Evaluation of an OAuth 2.0 protocol implementation for Web server applications," in *Proc. Int. Conf. Workshop Comput. Commun. (IEMCON)*, Oct. 2015, pp. 1–4.

[6] M. Noureddine and R. Bashroush, "A provisioning model towards OAuth 2.0 performance optimization," in *Proc. IEEE 10th Int. Conf. Cybern. Intell. Syst. (CIS)*, Sep. 2011, pp. 76–80.

[7] S. Emerson, Y.-K. Choi, D.-Y. Hwang, K.-S. Kim, and K.-H. Kim, "An OAuth based authentication mechanism for IoT networks," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2015, pp. 1072–1074.

[8] C. Ribeiro, H. Leitold, S. Esposito, and D. Mitzam, "STORK: A real, heterogeneous, large-scale eID management system," *Int. J. Inf. Secur.*, vol. 17, no. 5, pp. 569–585, Oct. 2018. doi: 10.1007/s10207-017-0385-x.

[9] J. L. Hernandez-Ardieta, J. Heppe, and J. F. Carvajal-Vion, "STORK: The European electronic identity interoperability platform," *IEEE Latin Amer. Trans.*, vol. 8, no. 2, pp. 190–193, Apr. 2010.

[10] H. Leitold and B. Zwattendorfer, *STORK: Architecture, Implementation and Pilots*. Berlin, Germany: Wiesbaden Vieweg Teubner, 2011, pp. 131–142.

[11] V. Koulolias, A. Kountzeris, H. Leitold, B. Zwattendorfer, A. Crespo, and M. Stern, "STORK e-privacy and security," in *Proc. 5th Int. Conf. Netw. Syst. Secur.*, Sep. 2011, pp. 234–238.

[12] D. Berbecaru, A. Atzeni, M. D. Benedictis, and P. Smiraglia, "Towards stronger data security in an eID management infrastructure," in *Proc. 25th Euromicro Int. Conf. Parallel, Distrib. Netw.-Based Process. (PDP)*, Mar. 2017, pp. 391–395.

[13] W. P. Filho, C. Ribeiro, and T. Zefferer, "Privacy-preserving attribute aggregation in eID federations," *Future Gener. Comput. Syst.*, vol. 92, pp. 1–16, Mar. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17327966

[14] S. Pearson, *Privacy, Security and Trust in Cloud Computing*. London, U.K.: Springer, 2013, pp. 3–42.

[15] C. Esposito, "Interoperable, dynamic and privacy-preserving access control for cloud data storage when integrating heterogeneous organizations," *J. Netw. Comput. Appl.*, vol. 108, pp. 124–136, Apr. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804518300316

[16] B. Zwattendorfer and A. Tauber, "Secure cross-cloud single sign-on (SSO) using eIDs," in *Proc. Int. Conf. Internet Technol. Secured Trans.*, Dec. 2012, pp. 150–155.

[17] T. Zefferer, D. Ziegler, and A. Reiter, "Best of two worlds: Secure cloud federations meet eIDAS," in *Proc. 12th Int. Conf. Internet Technol. Secured Trans. (ICITST)*, Dec. 2017, pp. 396–401.

[18] G. Aavik and R. Krimmer, "Integrating digital migrants: Solutions for cross-border identification from e-residency to eIDAS. a case study from Estonia," in *Electron. Government*, H. J. Scholl, O. Glassey, M. Janssen, B. Klievink, I. Lindgren, P. Parycek, E. Tambouris, M. A. Wimmer, T. Janowski, and D. Sá Soares, Eds. Cham, Switzerland: Springer, 2016, pp. 151–163.

[19] D. Hühnlein, "Towards eIDAS as a service," in *ISSE Securing Electronic Business Processes*, H. Reimer, N. Pohlmann, and W. Schneider, Eds. Wiesbaden, Germany: Springer, 2014, pp. 241–248.

[20] F. Jordan, H. Pujol, and D. Ruana, "Achieving the eIDAS vision through the mobile, social and cloud triad," in *ISSE Securing Electronic Business Processes*, H. Reimer, N. Pohlmann, and W. Schneider, Eds. Wiesbaden, Germany: Springer, 2014, pp. 81–93.

[21] G. Kaur and D. Aggarwal, "A survey paper on social sign-on protocol OAuth 2.0," *J. Eng. Comput. Appl. Sci.*, vol. 2, no. 6, pp. 93–96, 2013.

[22] D. Hardt, "The OAuth 2.0 authorization framework," Microsoft, Washington, DC, USA, Tech. Rep., Request Comments 6749, 2012.

[23] R. Boyd, *Getting Started with OAuth 2.0*. Newton, MA, USA: O'Reilly Media, 2012.

[24] B. Leiba, "OAuth Web authorization protocol," *IEEE Internet Comput.*, vol. 16, no. 1, pp. 74–77, Jan./Feb. 2012.

[25] F. Buccafurri, G. Lax, S. Nicolazzo, and A. Nocera, "A middleware to allow fine-grained access control of twitter applications," in *Proc. 2nd Int. Conf. Mobile, Secure, Program. Netw.* Paris, France: Springer, Jun. 2016, pp. 168–182.

[26] N. Naik and P. Jenkins, "Securing digital identities in the cloud by selecting an apposite federated identity management from SAML, OAuth and OpenID connect," in *Proc. 11th Int. Conf. Res. Challenges Inf. Sci. (RCIS)*, May 2017, pp. 163–174.

[27] J. Jensen, "Federated identity management challenges," in *Proc. 7th Int. Conf. Availability, Rel. Secur.*, Aug. 2012, pp. 230–235.

[28] J. Jensen, "Benefits of federated identity management-A survey from an integrated operations viewpoint," in *Availability, Reliability and Security for Business, Enterprise and Health Information Systems*, A. M. Tjoa, G. Quirchmayr, I. You, and L. Xu, Eds. Berlin, Germany: Springer, 2011, pp. 1–12.

[29] D. Fett and R. Küsters, and G. Schmitz, "A comprehensive formal security analysis of OAuth 2.0," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, New York, NY, USA, 2016, pp. 1204–1215. doi: 10.1145/2976749.2978385.

[30] W. Li and C. J. Mitchell, "Security issues in OAuth 2.0 SSO implementations," in *Proc. 17th Int. Conf. Inf. Secur.* Hong Kong: Springer, Oct. 2014, pp. 529–541.

[31] F. Yang and S. Manoharan, "A security analysis of the OAuth protocol," in *Proc. IEEE Pacific Rim Conf. Commun., Comput. Signal Process. (PACRIM)*. Victoria, BC, Canada, Aug. 2013, pp. 271–276. doi: 10.1109/PACRIM.2013.6625487.

[32] E. Shernan, H. Carter, D. Tian, P. Traynor, and K. Butler, "More guidelines than rules: CSRF vulnerabilities from noncompliant OAuth 2.0 implementations," in *Detection of Intrusions and Malware, and Vulnerability Assessment*. M. Almgren, V. Gulisano, and F. Maggi, Eds. Cham, Switzerland: Springer, 2015, pp. 239–260.

[33] S.-T. Sun and K. Beznosov, "The devil is in the (implementation) details: An empirical analysis of OAuth SSO systems," in *Proc. Conf. Comput. Commun. Secur.*, New York, NY, USA, 2012, pp. 378–390. doi: 10.1145/2382196.2382238.

[34] (2005).*SAML V2.0 Specification*. Accessed: Feb. 20, 2019. [Online]. Available: http://saml.xml.org/saml-specifications

[35] A. Alonso, F. Fernández, L. Marco, and J. Salvachúa, "Iaacaas: IoT application-scoped access control as a service," *Future Internet*, vol. 9, no. 4, p. 64, 2017. [Online]. Available: http://www.mdpi.com/1999-5903/9/4/64

[36] E. D. Hardt, *The OAuth 2.0 Authorization Framework*, document RFC 6749, Internet Requests for Comments, RFC Editor Oct. 2012.

[37] F. Turkmen and B. Crispo, "Performance evaluation of XACML PDP implementations," in *Proc. Workshop Secure Web Services*, 2008, pp. 37–44.

[38] L. Sánchez, V. Gutiérrez, J. A. Galache, P. Sotres, J. R. Santana, J. Casanueva, and L. Muñoz, "SmartSantander: Experimentation and service provision in the smart city," in *Proc. 16th Int. Symp. Wireless Pers. Multimedia Commun. (WPMC)*, Jun. 2013, pp. 1–6.

[39] J. Lanza, P. Sotres, and L. Sánchez, J. A. Galache, J. R. Santana, V. Gutiérrez, and L. Munoz, "Managing large amounts of data generated by a smart city Internet of Things deployment," *Int. J. Semantic Web Inf. Syst.*, vol. 12, no. 4, pp. 22–42, Oct. 2016. doi: 10.4018/IJSWIS.2016100102.

ÁLVARO ALONSO was born in Madrid, Spain, in 1988. He received the M.Sc. and Ph.D. degrees in telecommunication engineering from the Universidad Politécnica of Madrid, Spain, in 2012 and 2016, respectively, where he was a Research Assistant with the Next Generation Internet Research Group, from 2010 to 2018.

From 2016 to 2018, he was an Adjunct Professor with the Faculty of Economics and Business, Complutense University of Madrid. Since 2018, he has been an Assistant Professor with the Universidad Politécnica of Madrid. He is the author of several articles and participates in a lot of Spanish and European research projects. His research interests include multi-conferencing systems in cloud computing, security, and the IoT in the future Internet.

ALEJANDRO POZO was born in Ribadeo, Spain, in 1991. He received the M.Sc. degree in telecommunication engineering from the Universidad Politécnica of Madrid, Spain, in 2017, where he is currently pursuing the Ph.D. degree in telecommunication engineering and is a Research Assistant with the Next Generation Internet Research Group. He participates in several European projects and publishes results in journals related to the Internet of Things, security, and identity management.

JOHNNY CHOQUE received the degree in electronic engineering from the National University of Engineering, Peru, in 1995, and the M.Sc. degree (Hons.) and the Ph.D. degree *(cum laude)* in communications engineering from the University of Cantabria, Spain, in 1998 and 2014, respectively, where he has been a Researcher with the Communications Engineering Department, Since 2000, working in several projects of Framework Programme and Horizon 2020 of European Commission as well as in national and industrial projects. He has over 40 peer-reviewed publications in the field of next generation wireless networks and future Internet technologies. His current research interests include low power WAN, smart cities, the Internet of Things, and blockchain.

GLORIA BUENO finished her development of computer applications higher vocational training, in 2012. Since 2013, she has been with mashme.io. She holds several positions, such as Web Developer and IT Support Systems Engineer and currently as a Consulting Systems Engineer. She is being the bridge between Sales and IT teams with customer satisfaction as a goal.

JOAQUÍN SALVACHÚA was born in Madrid, Spain, in 1963. He received the M.Sc. and Ph.D. degrees in telecommunication engineering from the Universidad Politécnica Madrid, Spain, in 1989 and 1994, respectively, where he has been an Associate Professor, since 1995.

He is the author of several articles and participates in a lot of Spanish and European research projects. His research interests include advanced cloud and edge architectures, big data infrastructure, data privacy and usage control, NoSql databases, applications, and identity in blockchain, among other things.

LUIS DIEZ received the M.Sc. and Ph.D. degree from the University of Cantabria, in 2013 and 2018, respectively, where he is currently an Assistant Professor with the Communications Engineering Department. He has been involved in different international and industrial research projects. His research interests include future network architectures, resource management in wireless heterogeneous networks, and the IoT solutions and services.

JORGE MARÍN received the M.Sc. degree in telecommunications from the University of Sevilla, in 2011, where he is currently a Full-Stack Engineer and a Technical Architect, spending more than 15 years in very different environments, such as telecoms, banking, eLearning, and eCommerce. He is also a Software Developer with mashme.io, where he is involved in developing web-based videoconferencing systems.

PEDRO LUIS CHAS ALONSO graduated in telecommunications engineering from Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, in 1978. His professional career has taken place within the Telefónica Group, mainly in R&D activities related to broadband networks and services. From 1985 to 2007, he held several senior positions in Telefonica R&D, Telefónica Corporation, Telefónica de España, Telefónica Móviles, and Telefónica Soluciones. Since 2008, he has been a Contract Researcher with the Next Generation Internet Group of DIT/ETSIT/UPM. He has participated (and managed) many Research and Development projects both at Spanish and EU levels, related mainly to broadband networks and next generation Internet.