On enabling additional natural person and domain-specific attributes in the eIDAS network

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 14, 2021, accepted August 24, 2021, date of publication September 27, 2021, date of current version October 6, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3115853

On Enabling Additional Natural Person and Domain-Specific Attributes in the eIDAS Network

DIANA GRATIELA BERBECARU, ANTONIO LIOY, AND CESARE CAMERONI
Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy
Corresponding author: Diana Gratiela Berbecaru (diana.berbecaru@polito.it)

This work was supported by the European Union's Horizon 2020 Project "CyberSec4Europe" under Grant 830929.

ABSTRACT Within digital virtual space, secure and efficient user authentication and identification are essential to prevent identity theft and unauthorized access to sensitive information and services. The eIDAS network implementing the European Union (EU) Regulation 910/2014 links the electronic identity (eID) systems of EU countries to allow citizens' access by authenticating with government eIDs. At authentication time, the eIDAS nodes transfer core personal attributes (i.e., name, surname, date of birth, and an identifier) to the service providers (SPs). Since long-term applications require more personal or domain-specific data to provide the service or to perform identity matching, the SPs must obtain such data in an alternative way, with additional costs and risks. Herein, we extend the eIDAS network to retrieve and transfer additional person and domain-specific attributes besides the core ones. This process introduces technical, usability, and privacy issues that we analyze. We exploit a logical AP Connector between the eIDAS node and the entities providing additional attributes. We implemented two AP Connectors, named AP-Proxy and AP-OAuth2, integrated with the Italian pre-production eIDAS node to get additional attributes from the Politecnico di Torino university backend. In an experimental campaign, 30 students have accessed academic services at three foreign universities with recognized Italian eIDs, and transferred additional attributes over the eIDAS network. Despite some usability and privacy concerns encountered, the user experience was positive. We believe our work is helpful in the implementation of the recently adopted European Digital Identity framework, which proposes to extend the person identification data set recognized cross border, and the creation of digital wallets linking different data sets or credentials.

INDEX TERMS Electronic identity, eIDAS Regulation, digital identity management, attribute retrieval.

I. INTRODUCTION

It is common practice nowadays to use Internet for an increasing number of services. However, to securely access remote services in different domains (e.g., health, finance, or academia) strong and user-friendly authentication and identification is needed to prevent attacks, such as data leakage or identity theft. To this aim, since the late '90s, the European Member State (MS) countries have strengthened their eID management strategies.

An electronic (or digital) identity is a digital representation of a natural or legal person. To allow citizens to prove who they are (with high assurance level) in public or private services, the European governments have started to issue government eID credentials to citizens, such as national eID cards with a digital (public key) certificate on-board [1]–[3]. Such credentials are issued after completing various identification procedures, such as consulting national registries or identity documents. Therefore, they provide high assurance about a person's identity. Nowadays, more user-friendly credentials exist, such as one-time passwords used in combination with personal devices [4].

By exploiting these credentials, the citizens can authenticate at an Identity Provider (IdP) which may provide some core identification attributes about the citizen to access a Service Provider (SP). In this context, the IdP and SP establish a trust relationship via national digital identity systems, or through bilateral agreement. For example, in Italy, companies like InfoCert [5] or Poste Italiane [6] issue authentication credentials valid under the national system SPID (Sistema Pubblico di Identità Digitale, i.e. Public System for Digital Identity) [7]. The citizens may exploit these credentials to access SPID-enabled public or private services, such as registering children at school, or filing a tax return. In other

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

countries, the mobile industry played a significant role in building digital identity systems, such as in Estonia, Finland, Norway, and Switzerland [8].

To allow mutual recognition of eIDs and to support citizens' access to foreign services with national credentials, the European eIDAS (electronic IDentification, Authentication, and trust Services) network connecting the eID systems of several EU countries has been designed and implemented. This network follows the eIDAS Regulation 910/2014 [9] in permitting cross-border authentication with national credentials for both legal and natural persons [10]. It exploits national eIDAS nodes that communicate via the eIDAS protocol [11] based on SAML (Security Assertion Markup Language) 2.0. Inside each country, the eIDAS node interacts with the national identity infrastructure by using specific technologies and protocols. For natural persons, the eIDAS network transfers a restricted set of attributes named eIDAS Minimum Data Set (MDS) [11]. This set contains four mandatory attributes (`FamilyName`, `FirstName`, `DateOfBirth`, `PersonIdentifier`) and four optional ones (`BirthName`, `PlaceOfBirth`, `CurrentAddress`, `Gender`).

Motivation: Our work stems from the following research question: *Is the eIDAS MDS sufficient for the SPs to provide services (for natural persons) upon authentication through the eIDAS network?* As explained in Section IV, the answer to this question is not straightforward. In one-shot services, the SPs may exploit the eIDAS MDS attributes with no further requirement. On another hand, for long-term services the SPs need other citizen attributes in addition to the MDS ones. For example, they might need data about the nationality or photo for identity matching purposes [12]. Domain-specific data, like professional qualification, bank balance, medical or academic history, or current occupation, are helpful to provide specific services. Unfortunately, self-assessed user data cannot be considered trustworthy, so research is performed nowadays to collect such attributes from authoritative parties in a secure and privacy-preserving manner.

The eIDAS network could retrieve more data from trusted national sources. According to the eIDAS specification [11], the eIDAS nodes may support the exchange of other attributes. New attributes may be defined, but the ones not specified in the eIDAS attribute profile [10] may require a bilateral agreement. Put differently, countries may decide whether they want to exchange more attributes (in addition to the MDS) with other countries, and enhance the node for this purpose. The recently adopted European Digital Identity framework [13] published by the European Commission (EC) stresses the need to extend the eIDAS network with additional person identification data set recognised cross border to support identity matching. Moreover, this EC report proposes three options to underpin citizens and business to use eIDs together with electronic attestation of attributes and credentials linked to their eIDs. In practice, a trust service provider or a digital wallet could link the attributes and credentials to the eIDs of the users. In this work, we extend the eIDAS node (in the Specific part) to support additional attribute retrieval for natural persons. Moreover, we consider other issues, such as usability and privacy concerns.

Methodology: We have addressed several challenges in retrieving and transferring additional attributes over the eIDAS network:

1) Support for new attributes and for attribute retrieval on the eIDAS node. Deciding what kind of additional data (personal or domain-specific) and how much data the eIDAS network should transfer is an important topic. According to the privacy by default principle, the eIDAS nodes must limit the type of collected data (collection limitation) and retrieve just what is needed (data minimization) [14]. We selected a set of additional personal and academic attributes to enable on the eIDAS node. Then, we used two approaches, named AP Proxy and AP-OAuth2, to retrieve them from an Attribute Provider (AP) separate from the IdP(s) connected to the eIDAS network.

2) Design and implementation of AP Connectors. Authentication and attribute retrieval via the eIDAS network involve several entities, namely the user agent (browser), the SPs providing services, the eIDAS nodes, the IdPs authenticating citizens with national credentials, and potentially the APs providing additional person or domain-specific attributes. In this context, interoperability is a challenging task [15]. For user consent management and authorization, the local entities prefer to use "lighter" protocols like Open Authorization (OAuth) 2.0 [16], and simpler formats for attribute transfer, such as JSON [17]. To convert the eIDAS protocol messages to other specific protocols, we have implemented adapters in a logical AP Connector module.

3) User consent and privacy issues. Even when additional data is enabled in the eIDAS network, the user consent is required on the collected data. The user consent management needs attention because citizens must agree on the data retrieved and to whom it will be released. At the same time, the users must not perceive the user consent dialogs as (privacy) intrusive or repetitive. We analyze the user consent management, indicating different points in the eIDAS workflow where the consent is acquired. Moreover, in the proposed AP Connector implementations, the user consent is handled differently by the entities involved.

Contribution: Our main contributions are: (i) an in-depth analysis of the (natural person) attributes supported currently by the eIDAS network, and motivations for supporting new ones to respond to user identification and service needs; (ii) discussion on the exploitation of eIDAS MDS attributes in one-shot and long-term services; (iii) presentation of two AP Connector models for attribute retrieval employed by the eIDAS nodes; (iv) description of two possible AP Connector implementations and their testing in experimental testbeds. In the first one, named AP Proxy, the eIDAS node establishes a direct HTTPS backend channel for communication with the AP. In the second one, named AP-OAuth2, the node exploits the OAuth 2.0 protocol for the authorization of the released attributes. To validate the implementations,

we involved students that used their SPID authentication credentials to access Erasmus enrollment services at three foreign universities. Herein, we extend our previous work on integrating the APs into eIDAS [18]–[20].

Organization: The paper is organized as follows: Section II discusses the related work, while Section III brings details on the eIDAS network. Section IV analyzes the use of eIDAS MDS attributes in one-shot and long-term services, and explains the need to carry additional data in the eIDAS network. Section V presents the logical AP Connector and two eIDAS attribute enabling models, while Section VI describes the integration of the AP Connector with the Italian eIDAS node. Section VII explains the AP Connector implementations, as well as the experimental testbed involving the Italian eIDAS pre-production node and Politecnico di Torino infrastructure. Section VIII details the validation test sessions in which students have transferred specific data to perform enrollment at three foreign universities, as well as their feedbacks. Finally, Section IX concludes the paper and indicates future works.

II. RELATED WORK

In this section, we present first the basic concepts of the eIDAS network (Section II-A). Next, we select some journal or conference papers addressing the technical and user consent management issues in our work. In Section II-B, we review related work regarding privacy and user consent issues. Section II-C details previous work dealing with the user identification process, techniques for processing attributes, such as attribute aggregation and filtering, and trust models in digital identity management systems. Finally, Section II-D presents some related research projects.

A. THE eIDAS NETWORK: THE BASIS

The eIDAS Regulation supports the mutual cross-border recognition of government eIDs issued by schemes notified under eIDAS. The EU countries must recognize the eIDs of the countries that have *notified* their eID schemes. As explained in [13], the MS countries voluntarily prompt their eID scheme(s) to the European Commission, which involves MS experts to do a peer-review of the scheme, assessing its compliance with the criteria set out in the eIDAS Regulation, implementing acts and guidelines [21]. Following the notification and the completion of the peer-review process, the scheme is published on a special list of ''notified eID schemes'' [22]. For example, Italy has notified two schemes: the SPID system and the CIE (Carta d'Identità Elettronica).

To support the mutual eID recognition in practice, the eIDAS interoperability network composed of national ''eIDAS nodes'' has been set up and is operational nowadays. eIDAS defines three levels of assurance (LoA), namely `low`, `substantial`, and `high`, corresponding respectively to a limited, substantial, or high degree of confidence in the claimed or asserted identity of a person. The LoA level covers the methods used for identity proofing and credential issuance to citizens, as well as authentication mechanisms and credential management. The countries internally map the national authentication credentials of their eID schemes into the LoA levels. In general, the authentication with a national smart-card corresponds to a high LoA level. The SPs must accept citizen's authentication with substantial and high levels, and may accept the authentication performed with a low level. Further details on the eIDAS network architecture and protocol, the attributes supported and the underlying trust model, as well as an overview of some notified eID schemes are provided in Section III.

B. USER CONSENT AND PRIVACY ISSUES

Satchell *et al.* [23] discussed a relevant aspect in digital identity systems, that is the citizens want to be able to control and have the ability to create, maintain and share the information related to their eID(s). In general, they would like to remain ''anonymous'' during transactions and prefer to have multiple identities that may overlap.

Another study involving three famous web identity providers (Google, Facebook, and Google+) also indicates that the majority of the users (more precisely, 399 out of 424) said that it was ''very'' or ''extremely important'' for them to be in control over what data was passed by an IdP to an SP [24]. Furthermore, 50% of the participants in the above study preferred to use multiple IdPs instead of a single one. The same study mentions another worthy aspect: the participants didn't have a precise understanding of what data has been sent to the SPs. Nevertheless, when more data has been transferred, they did realize that more attributes have been sent, even if not which ones. Their natural tendency to ''log in'' was not significantly affected by the consent dialogs but by the privacy concerns. For example, most of the people felt ''uncomfortable'' to send their friends list and photos, no matter if they were sent to a trusted or an untrusted website.

Thus, when more data passes through the eIDAS network, the selection of requested attributes needs proper attention. The attributes mandatory for the service must be clearly marked and explained. If too much data is asked, the persons might become skeptical about using the platform due to privacy concerns. Gomi proposed a framework for tracing history of identity information transfers across different domains, allowing user control over personal information propagation [25]. Other approaches, such as My Data [26], put the person at the center of the use of data. MyData promotes awareness about the relevance of personal data and its more ethical use. They encourage the consciousness of both users and companies, suggesting technical principles as well. The recent EC report [27] analyzes how the eIDAS Regulation supports the requirements for customer data portability and gives indications on the control of credentials by the user through the verifiable claims that are emerging in a number of initiatives, such as the European Blockchain Service Infrastructure. Taniguchi *et al.* in [28] proposed a scheme to protect digital identities' privacy by using anonymity and pseudonymity. The idea is that under ordinary circumstances, a person can act anonymously, but in

TABLE 1. Referenced papers.

Reference	Summary	Topics: 1. Privacy 2. User Consent 3. Multiple IDs, anonymity, pseudonymity 4. Attributes transfer and mapping			
		1	2	3	4
[23]	Discusses users' needs to create and control their digital representation. Beyond "security" and "authentication" it's important to include "user control" and "portability" in the design of digital identity systems.	✓	✓	✓	✗
[30]	Analyzes different models of attribute aggregation in federated identity management. Three models are detailed: aggregation at the SP, aggregation at the IdP, aggregation at the client.	✓	✗	✗	✓
[28]	Describes how to protect digital identities' privacy by using anonymity and pseudonymity.	✓	✗	✓	✗
[31]	Describes trust requirements of different identity management models: isolated, federated, centralised, or personal.	✓	✗	✗	✓
[29]	Introduces a privacy-preserving solution for digital identities management in federated systems.	✓	✗	✗	✓
[25]	Proposes a framework for tracing history of identity information transfers across different domains, allowing user control over personal information propagation.	✓	✗	✓	✗
[39]	Reviews the support for attributes in the STORK, STORK 2.0 and eIDAS infrastructures, as well as some approaches adopted for attribute retrieval and transfer.	✗	✓	✗	✓
[19]	Describes the design and implementation of two services (Login and Wi-Fi access) exploiting the eIDAS network and the eIDAS MDS attributes.	✓	✓	✗	✓
[18]	Describes the extension of the eIDAS network with support for additional attributes in the academic domain. It introduces the AP Connector logical module.	✗	✗	✗	✓

case of a special situation, the identity of the involved person can be specified.

C. TECHNICAL ISSUES

1) USER IDENTIFIER(S) AND IDENTITIES

Bhargav-Spantzel *et al.* [29] classified the user identifiers as *weak* and *strong*. A strong identifier uniquely identifies an individual in a population, while a weak identifier can be applied to many individuals in a population. Whether an identifier is strong or weak depends upon the population size and the uniqueness of the identifying attribute. The same authors observed that multiple weak identifiers may lead to a unique identification. This work is useful in the scope of this paper because it indicates that several attributes are typically needed for person identification. Berbecaru *et al.* discussed the problem of homonyms (persons with the same name, surname, date of birth) in the person identification process [19]. The same authors explained that the eIDAS MDS attributes carrying the name, surname, and date of birth are not sufficient to distinguish a person from another one, while the eIDAS PersonIdentifier could not be meaningful for the SP. For this reason, in [19], the authors adopted a solution in which the user was asked to bind his *current* eIDAS PersonIdentifier to his national unique identifier. Moreover, the EC report [13] notes that "the rigid data set for notified eID makes it also difficult to match identity records as the current minimum data set (eIDAS MDS) is often not sufficient to uniquely identify a person".

2) ATTRIBUTE AGGREGATION

Ferdous and Poet [30] analyzed different models of attribute aggregation in federated identity management systems. They discussed the trust requirements considering the classical actors in the federated model (SP, IdP, and the client), and they modeled attribute aggregation at each side. We note the "identity proxying model" in which the SP allows the user to aggregate attributes from multiple IdPs using a highly trusted IdP (a kind of "super IdP"). In this model, the user is first redirected to the trusted IdP, which subsequently forwards the user to other multiple IdPs. After the user is authenticated separately at each IdP, the user returns to the trusted IdP with an assertion containing the assessed attributes. At last, the trusted IdP validates each assertion, retrieves the attribute values, and combines them. The trusted IdP may add its user attributes to the combined set and then reasserts all attributes to the SP. The SP has a trust relationship with the trusted IdP, and it is not aware of the other IdPs from where the attributes have been collected. There are some similarities between the Italian IdP Proxy (part of the eIDAS node) described in this paper and the trusted IdP in [30]. They both act as attribute collector and aggregator, trusted by the user. In contrast, the authentication (through eIDAS) is performed by the user just once at an IdPs implementing a recognized eID scheme, and the SP establishes a trust relationship with the local eIDAS node. Further details on the eIDAS trust model are presented in Section III.

3) TRUST MODELS

Jøsang *et al.* [31] analyzed trust requirements of different identity management solutions in a simplified model composed of clients, service providers, and identifier & credential providers. Depending on the interactions among the above entities, the authors divided the architectures in 4 types: isolated, federated, centralised, or personal. In each

architecture, the main trust requirements have been sketched. Compared to the previous models, the eIDAS network adds an additional element, namely the eIDAS node, which actively communicates with the other nodes and the national entities. More details on the eIDAS trust model are presented in Section III-D.

D. RELATED PROJECTS

Some research projects, such as the eID4U project [32] or the DE4A project [33], tried to endorse more attributes into the eIDAS network. They proposed an enhancement of the eIDAS network by adding new personal and academic attributes to support user identification and new eIDAS-enabled services [34].

In contrast, other projects such as [35], [36] have studied how to integrate eIDAS authentication into other domain-specific platforms spanning multiple countries. For example, in the famous eduGAIN network, which implements a federation of universities belonging to different countries across the globe, the user authenticates with his university authentication credential(s) to exchange academic attributes. The MyAcademicID project [37] has designed and implemented a solution to support eIDAS authentication in eduGAIN services. To distinguish users inside the federation, the project defined a new academic identifier (named ''European Student Identifier'') different from the eIDAS identifier. Moreover, to convert the messages from the eIDAS format into the format recognized by the eduGAIN network, the project has designed and implemented a dedicated bridge.

The recent mGov4EU project [38] instead will combine mobile authentication and cross-border authentication through the eIDAS network with other platforms to create a trustworthy federation. The final goal is to allow citizens' access to cross-border services in a privacy-preserving manner by exploiting modern smartphones, and by supporting user consent management via user-controlled release of authoritative data.

III. MORE ON THE eIDAS NETWORK

In this section, we present the characteristics of the eIDAS network, the trust model, as well as the architectural components. Moreover, we indicate the code implementing the eIDAS specification and the deployment environments.

Initially prototyped in the frame of STORK and STORK 2.0 [39], [40] European-funded projects, the eIDAS network is nowadays the de facto Pan-European eID interoperability framework connecting the digital identity systems of several EU countries to allow the integration of e-services in various domains [41].

We provide clarifications about the eIDAS node implementation, in particular about the *Generic* and the (MS) *Specific* parts used to connect the node to the national entities. The Generic part contains modules (common to all the eIDAS nodes) used for cryptographic processing of the SAML metadata and the eIDAS authentication request and response messages. The Specific part contains modules (specific for each country) allowing the communication of the eIDAS node with the local IdPs and SPs. Moreover, this part may be further extended for the interaction with the local APs.

A. eIDAS NETWORK CHARACTERISTICS

The eIDAS network responds to several essential requirements, such as *decentralization* and *security*. Moreover, it could become in the future more *cross-sectorial*. We discuss these characteristics further below.

1) DECENTRALIZATION

There is no central (control or data storage) point in the eIDAS network. The authentication requests and responses are transferred through dedicated (national) eIDAS nodes, which are in a circle of trust. The nodes bilaterally exchange SAML metadata for the trust establishment, named eIDAS metadata throughout the paper.

2) SECURITY

The security of the eIDAS network as a whole is not trivial, given its decentralized nature and that different parties run the various components. The operators of the eIDAS nodes (national agencies or public ministries) must follow strict cryptographic requirements [42] for eIDAS message protection and TLS channel creation to avoid some possible attacks. For example, in the TLS connections [43], the eIDAS nodes must use cipher suites with forward secrecy and must use qualified X.509 certificates.

The data transferred through the eIDAS network has to be adequately protected and processed [44]. As noted in [29], ''the security and privacy of the user identity information, both certified and uncertified, are of utmost importance today.'' Security prevents theft and impersonation when the natural person attributes are retrieved and transferred through the eIDAS nodes, while privacy protects against the attributes disclosure. According to the eIDAS specification, ''Node operators of eIDAS nodes shall prove that …the node fulfills the requirements of standard ISO/IEC 27001 by certification, or by equivalent methods of assessment, or by complying with national legislation.'' Moreover, for privacy reasons, ''Nodes must not store any transaction data containing personal data beyond as required by Article 9(3) of [45]''. In brief, the eIDAS node operator shall store for a period, according to national requirements, (only) the data which, in the event of an incident, enable reconstruction of the sequence of the message exchange for establishing the place and the nature of the incident. This data is composed of: a) eIDAS node's identification, b) a message identification, and c) the message date and time. While indications are given for the security of the messages exchanged with the other eIDAS nodes, the protection of data exchanged with the national SPs and IdPs is country specific.

3) CROSS-SECTORIAL

The cross-sectorial feature means that attributes in one (specific) domain provided through the eIDAS network could

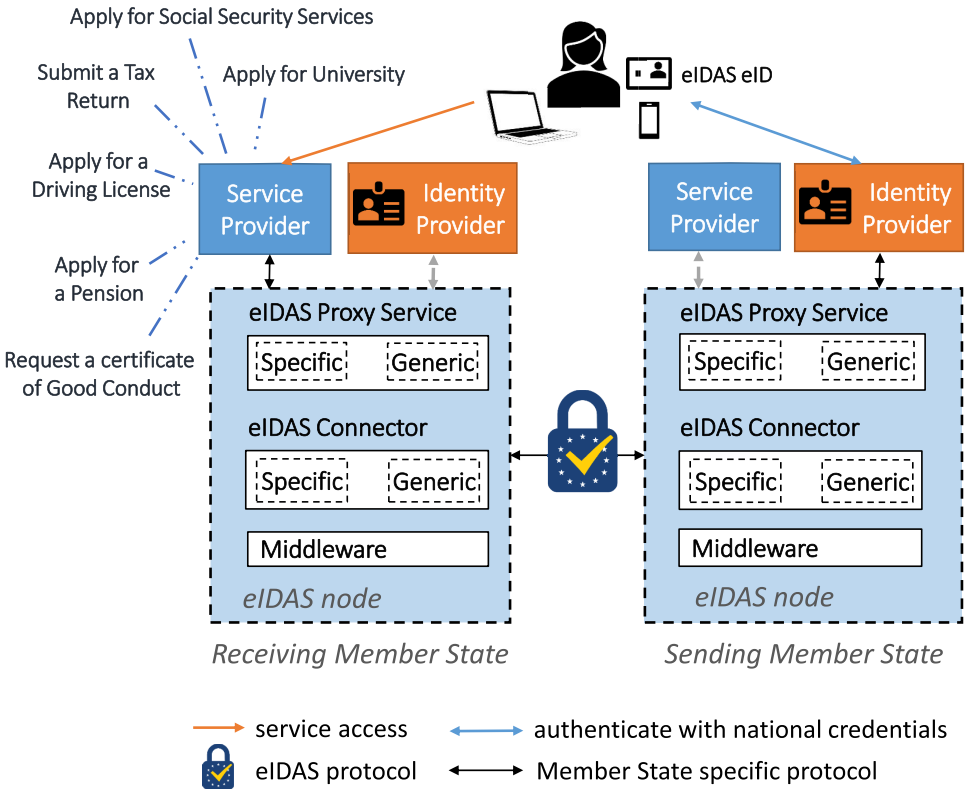

FIGURE 1. eIDAS nodes of the network, with the generic and specific parts. Citizens access eIDAS-enabled services with recognized national eIDs.

be exploited in other application domains. For example, the academic status of a person (e.g., student) could be used in public transport domain services to obtain personalized discounts.

B. eIDAS NODE COMPONENTS AND ATTRIBUTES

The eIDAS node is composed of two main logical entities, the eIDAS Connector and the eIDAS Proxy Service (shown in Fig. 1). The countries host one Proxy Service and one or more eIDAS Connector(s) providing public or private services. Instead of a Proxy Service, Germany provides a *Middleware* software to be installed and run on the other MS countries' eIDAS nodes. The (eIDAS) Connector in the so-called Receiving MS country receives the eIDAS authentication request (*eIDAS-Auth-Req*) from the national service provider and forwards it to the foreign eIDAS Proxy Service in the Sending MS country. The eIDAS Proxy Service processes the *eIDAS-Auth-Req*, typically by converting it into a request (in local format) sent to the national IdP(s) via MS-specific protocols. Upon successful authentication with eIDs recognized under eIDAS, the IdP generates an authentication response in local format and sends it back to the local Proxy Service. Based on the received response, the Proxy Service generates an eIDAS authentication response (*eIDAS-Auth-Res*) and sends it to the foreign Connector, which further processes it. The Connector may return the *eIDAS-Auth-Res* in eIDAS format (if the SP supports the eIDAS protocol), or it converts the response into the specific protocol supported by the SP. More in detail, both the Connector and the Proxy Service are composed of a Generic part and a Specific part. The Generic part allows the communication with the counterpart nodes through the eIDAS protocol [11]. The Specific part is involved in the communication of the eIDAS node with the national SP(s) and IdPs.

The *eIDAS-Auth-Req* carries, among other things, the list of requested attributes. The eIDAS attribute profile [10] defines eight MDS attributes for natural persons (see Table 2): four of them are mandatory, that is `PersonIdentifier`, `FamilyName`, `FirstName`, and `DateOfBirth`, and the other four attributes are optional (i.e., `BirthName`, `PlaceOfBirth`, `CurrentAddress` and `Gender`). Thus, only the minimum trustworthy personal data (something like a core data set) [46] must be transferred over the eIDAS network. In Section IV, we analyze these attributes more in depth, in particular the eIDAS `PersonIdentifier`.

C. CODE VERSIONS AND RUNNING ENVIRONMENTS

The operational eIDAS nodes must follow the eIDAS specification [11]. Some countries have developed their ad-hoc implementation, while others are running the eIDAS code(s) released by the European Commission adapted in the Specific part. Two eIDAS code versions (both in Java) have been released, namely the eIDAS code version 1.4.x (branch) and

TABLE 2. eIDAS natural person attributes. Mandatory attributes are marked with an asterisk.

FriendlyName	Description	Type	Example
PersonIdentifier*	Unique identifier. The first part is the ISO 3166-1 alpha-2 code of the IdP country followed by a slash, the second part is the ISO 3166-1 alpha-2 code of the SP country followed by a slash, the last part is a combination of readable characters	xsd:string	IT/ES/14AHSFFD56
FamilyName*	Current Family Name	xsd:string	Rossi
FirstName*	Current First Name	xsd:string	Marco
DateOfBirth*	Date of Birth. A date following the format YYYY-MM-DD	xsd:date	1980-11-05
BirthName	First Names at Birth and/or Family Name at Birth as a single text value	xsd:string	Marco Antonio Rossi
PlaceOfBirth	Place of Birth	xsd:string	Abbiategrasso
CurrentAddress	Current Address. It can be a single XML tag containing the full address or a sequence of XML tags containing different elements of the address	xsd:string	<eidas:FullCvaddress>Via Listz 21 00144 Roma</eidas:FullCvaddress> or <eidas:LocatorDesignator>21</eidas:LocatorDesignator> <eidas:Thoroughfare>Via Listz</eidas:Thoroughfare> <eidas:PostName>Roma</eidas:PostName> <eidas:PostCode>00144</eidas:Postcode>
Gender	Gender. Values accepted: Male, Female or Not Specified	xsd:string	Male

the eIDAS code version 2.x (branch). Since each release contains enhancements, we will detail (only) the main differences between the two code branches.

In the eIDAS code version 1.4.x, the Specific components translate the messages between the national eID scheme's format and the eIDAS one. The communication with the Generic components is done via the eIDAS protocol. In this version, the Generic and Specific parts are independent services that may even run on separate servers. In the eIDAS code version 2.x instead, the translation of the authentication messages is split between the two components: the Specific part translates between the national eID scheme and an intermediate format, and the Generic part translates between the intermediate format and the eIDAS one. In the eIDAS 2.x version, the communication between the Specific and the Generic part takes place by using a so-called lightweight protocol, and the two components are supposed to run on the same server.

The eIDAS code is usually deployed in different environments. The official nodes are available in *production* environments, while the experimental ones are running in *test* environments. The *pre-production* or Quality Assurance (QA) environment is normally used to run tests on a near-production environment. Many countries run the eIDAS code in production and most of them have dedicated test and pre-production environments. We have tested the support for additional natural person attributes in the eIDAS node in pre-production environment. The eIDAS nodes running in production environment support for now only the eIDAS MDS attributes.

D. TRUST MANAGEMENT

The eIDAS nodes are securely identified through eIDAS metadata exchange to provide an uninterrupted chain of trust [47]. Each MS is trust anchor for its own eIDAS node, that is there is no central trust anchor for all Member States, e.g., at European Commission site. The trust anchors are exchanged bilaterally between MS countries. Consequently, each MS country running an eIDAS node securely distributes the eIDAS metadata signing certificate ($Cert_{eIDAS\_metadata\_sign}$) that will be used by the other MS countries to verify the signature on its eIDAS metadata file. The mechanisms used for eIDAS metadata exchange and verification, as well as for pre-fetching and caching of metadata, are described in [47].

The eIDAS metadata file contains the X.509 digital certificates required by the other nodes for the verification of the signatures on the *eIDAS-Auth-Req* and *eIDAS-Auth-Res* messages, and for the decryption of the attribute values contained in the *eIDAS-Auth-Res* messages. In more detail, the eIDAS metadata of each node contains:

- the certificate ($Cert_{eIDAS\_message\_sign}$) needed to verify the digital signatures on the *eIDAS-Auth-Req* and *eIDAS-Auth-Res* messages generated by that eIDAS node.
- the certificate ($Cert_{eIDAS\_attributes\_encrypt}$) employed to encrypt user attributes contained in *eIDAS-Auth-Res* messages sent to the other eIDAS nodes.

E. OVERVIEW OF SOME NOTIFIED eID SCHEMES

Nowadays, 19 notified eID schemes coming from 15 different MS countries can be used for cross-border authentication and identification in eIDAS [48]. In this section, we indicate some of these schemes, while a full list of schemes along with the LoA levels of the credentials accepted under eIDAS is provided in [49]. Note that some countries (such as Austria, or Slovenia) have not notified their schemes, for diverse reasons explained in [13]. This fact implies that the other countries may accept authentication with the eIDs of non-notified scheme(s), but there is no obligation.

In Belgium, the deployment of the first generation of eID cards started in 2003. In 2014, the second generation of eID cards have been deployed reaching about 2 million cards delivered per year [50]. In 2018, there was an eID card reader in 49% of households [51], while in 2020 more than 2.5 million people used the mobile-based ''itsme'' authentication system [52]. In the Czech Republic, holders of the national eID can use it to access health insurance companies, online gaming, betting websites, and a law firm [13]. In Denmark, the NemID scheme used for authentication in online banking is exploited by about 4.7 million citizens, generating more than 55 million transactions per month [53]. In Estonia, 98% of Estonians have a national eID card and 67% of them use it regularly [54]. In particular, in 2018, 60% of the eID card owners used it at least once for authentication or signing purposes [55]. In Germany, 53 million eID cards were issued until 2018, but the aim is to reach 100% of the eligible population by 2020 [1]. In Italy, more than 19.5 million citizens have an eID card (CIE - Carta d'Identità Elettronica) [2]. The SPID credentials have grown exponentially, reaching more than 18 million issued credentials in a short time [56]. In Latvia, the eID cards are optional but they will become mandatory by 2023 [57]. Also in Luxembourg the eID cards are optional and are proposed to ID card applicants [58]. Portugal has started the deployment of the national eID cards (Cartão de Citadão) since 2008, and about 45% of cardholders have activated the digital certificate required for authentication and signature [3]. However, nowadays, the mobile authentication and signature solution named ''Chave Mòvel Digital'' is increasingly used, reaching 160,000 users in 2018 [4]. In Slovakia, eID cards deployment started in December 2013 [59] reaching 600,000 authentications per month in 2019 [60]. Spain deployed the eID cards in 2006 (DNIe - Document Nacional de Identidad electrónico), and in 2015 initiated the roll out of the DNIe 3.0, which added NFC support [61]. In the Netherlands the DigiD authentication solution had 13.8 million active accounts in 2018, generating a total of more than 307 million authentications [62].

IV. eID IDENTIFIERS AND SERVICES

A. NATURAL PERSON, DIGITAL IDENTITIES, eIDAS IDENTIFIERS

What is a digital identity? Although several definitions exist for the term ''digital identity'', we have considered in our work the ones provided by ITU-T and by the eIDAS Regulation.

The ITU-T Focus Group on identity and authentication defines digital identity as ''mechanisms that assert and verify personal data attributes in the context of digital services and transactions, based on three processes: identification, authentication, and authorization'' [8]. The ITU-T Group defined three digital identity types [8]: *foundational*, *functional*, and *tranactional*. A foundational (core) digital identity is created as part of a national digital identity scheme or similar, and is based on the ''formal establishment of identity through the examination of qualifying (breeder) documents such as birth records, marriage certificates, and social security documents''. The functional digital identity addresses the specific needs of an individual sector, such as healthcare. The transactional identity is ''intended to ease the conduct of financial or other transactions across multiple sectors''. The same report notes that state-issued eID acts as a strong, reliable foundational identity. Moreover, it also indicates that the digital identity for one person can be defined by two types of attributes: *biographic* attributes, such as name, age, gender; and *biometric* attributes, such as fingerprints, iris texture, voice, or facial geometry. Biometric attributes are crucial to uniquely identify a person when civil registration systems lack, or when official birth certificates are absent, as it happens in the developing countries [63].

The eIDAS Regulation defines the digital ID as ''the process of using personal identification data in electronic form uniquely representing either a natural or legal person, or a natural person representing a legal person''. For the natural persons, eIDAS deals with biographic attributes, because the national eID schemes get the person identification data from national registries, or official (identification) documents. Based on the above definitions, we conclude that the eIDAS network deals with the transfer of biographic attributes belonging to foundational digital identity.

Discussion on eIDAS `PersonIdentifier`*:* The eIDAS MDS attributes (except for the eIDAS PersonIdentifier) are part of the foundational digital identity of a person. What about the eIDAS `PersonIdentifier`? This eIDAS attribute has a specific format, which is described briefly in Table 3 [10]. It may be derived from the national identifier, but this is not mandatory since every MS country decides how to create it, except for the nationality codes. It might even be a pseudonym, which changes for every transaction [46]. A natural person may have several eIDAS PersonIdentifiers. These identifiers are guaranteed to be unique, in the sense that no two persons across the EU may have the same eIDAS PersonIdentifier. We observe that the national identifier of a person is not typically his eIDAS `PersonIdentifier`.

TABLE 3. Description of the eIDAS PersonIdentifier attribute.

Description	Example[(1)]
Nationality Code of the (citizen) identifying country, followed by a slash	IT/
Nationality Code of the service providing country, followed by a slash	AT/
A combination of readable characters that uniquely identifies the asserted identity in the country of origin. It does not necessarily reveal subject's actual identifier, like username or fiscal number	02635542Y

[(1)]Example of an Italian eID for an Austrian SP: IT/AT/02635542Y

When a citizen presents his eIDAS PersonIdentifier (along with his name, surname, and date of birth) to a foreign SP, additional data might be needed to provide the service. In long-term services, the SP may ask the citizen to register into a national registry, or provide additional personal data. In the

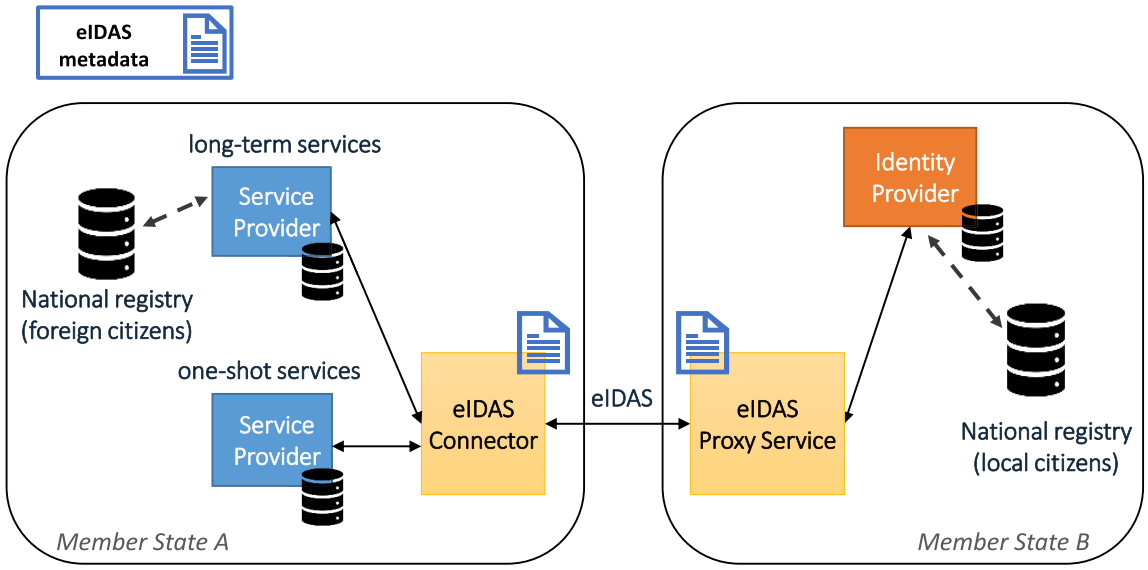

FIGURE 2. Exploiting the eIDAS network in one-shot and long-term services.

short term (or one-shot) services instead, the eIDAS MDS attributes may be considered sufficient to provide the service (as described in Section IV-B).

From the functional point of view, the SPs ask for a single, unique, and (possibly) a permanent person identifier for every European citizen, similar to the ones used at national levels. Although the eIDAS `PersonIdentifier` does not fully respond to all SP expectations, it has been considered as an acceptable choice because each MS country can choose its value, as long as it remains unique across the EU. Some possible options are: a) the eIDAS `PersonIdentifier` is derived from the national identifier, so its value would be as persistent as the national identifier; b) it is assigned/derived by the national IdP(s) performing the eIDAS authentication; c) it is an eID *pseudonym* changing from one transaction to another.

B. SERVICES EXPLOITING eIDAS

Some services do not request strict identification of a person, while other services do. For example, accountability requires the ability to uniquely identify the natural person. We distinguish the following types of services: one-shot and long-term. For each service type, we discuss the requirements in terms of the eIDAS attributes needed.

One-Shot Services: These services (e.g., bike renting, ski pass registration) are expected to last for a short amount of time. In such services, the eIDAS MDS attributes should be sufficient to provide the service. For example, in case of bike-sharing service, the authentication through the eIDAS network may allow a foreign citizen to reserve the bike, because the SP may store the eIDAS MDS attributes and provide the service. If discounts are applied based on age (e.g. for skipass cards issuance), the certified attribute containing the date of birth is sufficient. In case of a dispute (e.g., in case of bike damage), the SP might ask for further information about the citizen's identity in his country of origin, based on the eIDAS MDS attributes. Note that the entity that assigned the eIDAS `PersonIdentifier` may resolve possible homonyms, for example by querying its local database or a national registry of local citizens (if this is supported and allowed) as shown in Fig. 2.

Long-Term Services: In these services, the eIDAS MDS attributes provide core trustworthy data about a person. Anyhow, the SP typically needs more data for the service or has to implement additional checks or mechanisms for identification before providing the service. For example, to file a tax return in an eIDAS-enabled service, the Tax Agency might require the foreign citizen to register on the dedicated web portal. As part of this process, the Tax Agency would create an identifier (in the national format) for the foreign citizen, and the eIDAS `PersonIdentifier` along with the other attributes retrieved through eIDAS network would be part of the newly created profile. The whole data might be stored in a national registry (for foreign citizens), as shown in Fig. 2. The meaningful information in the service is the national identifier assigned to the foreign citizen. If needed, the Tax Agency would ask for additional information about the person with *that* eIDAS `PersonIdentifier` in the citizen's country of origin. However, for the moment, there is no automatic procedure for this purpose.

Another example long-term service is the "Login with eIDAS" service for academic personnel [19]. In this case, the university (acting as SP) has already registered the persons and indexed them in the internal database based on their national identifier. To allow eIDAS authentication, the person needs to bind in his profile the *current* eIDAS PersonIdentifier to his national identifier. If the eIDAS PersonIdentifier changes (as happens for example in Italy if

a person authenticates with different SPID IdPs), then the person has to update the identifier into his profile otherwise the eIDAS authentication fails.

We observe that in long-term services, the SPs typically call for *identity matching* [12]. This means they need to check whether the person authenticated through the eIDAS network matches one of the persons already registered on their side. To this aim, they could confront the eIDAS MDS data with the data/records kept locally in the SP's database or retrieved from a centralized national source like the national civil registries. Nevertheless, this process is subject to false positives or false negatives due to homonyms or transliteration problems. Homonyms occur when different persons have identical name, surname, and date of birth. Furthermore, transliteration problems, (i.e., minor differences encountered in the registered names or surnames) could require manual checking by the SP operator. This situation occurs especially in case of names and surnames containing accented or special characters. To improve the identity matching at the SP, one solution could be to extend the eIDAS MDS attributes to hold other identification data, such as the passport or the eID card number, a photo, or the European Health Insurance Card number.

V. CONNECTING ATTRIBUTE PROVIDERS TO THE eIDAS NETWORK

In general, services need additional data about citizens, such as profession (teacher, doctor), the role inside an organization (manager, director), or nationality. The eIDAS node may retrieve such attributes from a dedicated AP, or from an IdP acting as an AP. In our work, we consider only the APs in the citizen's country in which he has authenticated with his eIDAS eID. To obtain additional trustworthy attributes from other sources, the following questions arise:

Q1: Which entity is the Source of Authority for a specific attribute?

Q2: Which protocol should be used to get the additional attributes from the APs?

Q3: How to map the attributes from AP/national format into the format recognized by the eIDAS network?

Q4: Where and how should the user provide consent for the data to be transferred?

Moreover, the APs might need to perform *identity matching* as well. Since the eIDAS MDS attributes alone might not be sufficient because homonyms and transliteration problems might occur, the AP needs other attributes to avoid situations in which a person is mapped to someone else's profile, or is denied access to his profile. Typically, the AP expects information like the unique national identifier, a Tax Reference number, a Passport or ID card number, of a combination of them.

A. ATTRIBUTES CLASSIFICATION

In our work, we addressed attributes classification because the eIDAS nodes potentially retrieve attributes from different APs. We divide the attributes properties into two main categories: (1) general properties, that describe inherent characteristics of the attributes; and (2) attribute value properties that hold details about attribute assessment.

The *general properties* of the attributes are for example: (a) the Category, which can be personal or sector-specific; (b) the Persistence (permanent, non permanent), depending on whether its value may change in time (for the same person); (c) Strength (strong or weak, where a strong attribute can uniquely identify a user while many different weak attributes may be necessary at the same time to reduce the risk of wrong identification).

For example the Italian fiscal number (called also ''tax reference number'', or ''codice fiscale'') is personal, permanent and strong, while the passport number is personal, non permanent and strong. *Attribute value properties* are for example the Attribute Level of Assurance (ALOA), or Source of Authority (SoA). In the future, different ALOA levels (low, medium or high) could be defined for the level of trustworthiness of the attribute values. The SoA instead provides information about the entity (organization) retained liable for an attribute value. In our view, the SoA is composed of several subparts, such as the name of the entity or authority, and the URI which can be used to retrieve the attribute value.

B. AP CONNECTOR MODELS

In the eID4U project, we defined new attributes (shown in Table 4), and a logical AP Connector component to retrieve the additional attributes from the APs [34]. Different entities can run this component, which communicates with the eIDAS nodes and with the national APs by exploiting different technologies.

Two AP Connector models have emerged in the eID4U project. In the first one, the AP Connector interacts with the Specific part of the eIDAS Proxy Service to perform attribute processing, such as the attribute retrieval, the filtering or aggregation with other attributes valued by the IdP, and the conversion into eIDAS format. In this model, the AP Connector can be a stand-alone element outside the node, as depicted in Fig. 3, or it can be a component of the Specific part of the eIDAS Proxy Service. In a variant of this model, the AP Connector interacts directly with the eIDAS Proxy Service, and not only with its Specific part. This solution can be adopted by MS countries that have designed and implemented their own code for the eIDAS node. Thus, they might not have separated parts for the Generic component and the Specific one(s).

In the second model, the eIDAS node is only slightly modified to support new attributes and to transfer them to the counterpart nodes, but it does not either collect attributes or map the attributes to/from eIDAS format to national formats. In this case, the eIDAS node is agnostic about the attribute names (formats) used inside the country, and it does not convert the messages into other specific formats. The above tasks are performed mainly by the IdP, which retrieves additional attributes from one or more APs. In this case, the AP Connector is placed in between the IdP and the AP,

TABLE 4. Additional eIDAS attributes defined in eID4U project (personal and specific in the academic domain).

eIDAS Attributes	Description	Type
`TaxReference`	See `fiscalNumber` in Table 5	
`IdType`	One of: National Identity Card, Passport	xs:string
`IdNumber`	Document ID number	xs:string
`IdIssuer`	Document issuer	xs:string
`IdExpiryDate`	Date in the format YYYY-MM-DD	xs:date
`EhicId`	European Health Insurance Card ID. The value is composed of 20 digits starting with 80	xs:string
`Nationality`	ISO 3166-1 alpha-2 code of the country	xs:string
`Citizenship`	ISO 3166-1 alpha-2 code of the country	xs:string
`MaritalState`	One of: Single, Married, Divorced, Widowed, Civil Union	xs:string
`CountryOfBirth`	ISO 3166-1 alpha-2 code of the country	xs:string
`CurrentPhoto`	Base64 encoded picture binary	eid4u:document
`TemporaryAddress`	See `CurrentAddress` attribute in Table 2	
`Email`	Email address	xs:string
`Phone`	Phone number	xs:string
`HomeInstitutionName`	Name of the home institution	xs:string
`HomeInstitutionIdentifier`	Erasmus code of the home institution	xs:string
`HomeInstitutionCountry`	ISO 3166-1 alpha-2 code of the country	xs:string
`HomeInstitutionAddress`	See `CurrentAddress` attribute in Table 2	
`CurrentLevelOfStudy`	ISCED Code representing the level of current study	xs:integer
`FieldOfStudy`	ISCED Code representing the field of study	xs:integer
`CurrentDegree`	Name of the degree the user is attending at home institution	xs:string
`Degree`	ISCED Code representing previously achieved level of study	xs:integer
`DegreeAwardingInstitution`	Name of the degree awarding institution	xs:string
`GraduationYear`	Graduation year	xs:integer
`DegreeCountry`	ISO 3166-1 alpha-2 code of the country	xs:string
`LanguageProficiency`	Base64 encoded Europass 3.3 compliant declaration of language proficiency	europass3: ForeignLanguageSkillType
`LanguageCertificates`	Base64 encoded list of documents binaries (e.g. PDF files)	eid4u:document

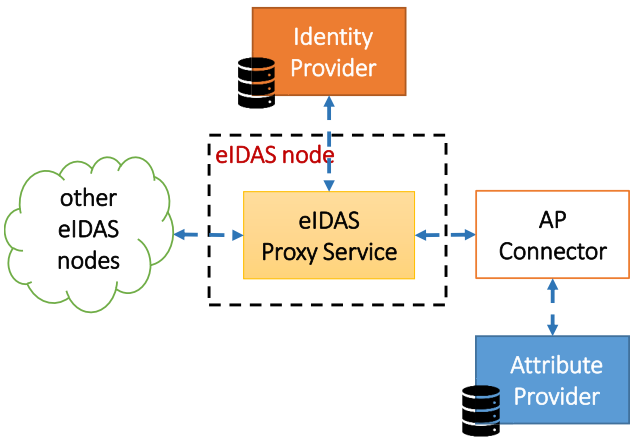

FIGURE 3. AP connector interacting with the eIDAS Proxy service and the attribute provider.

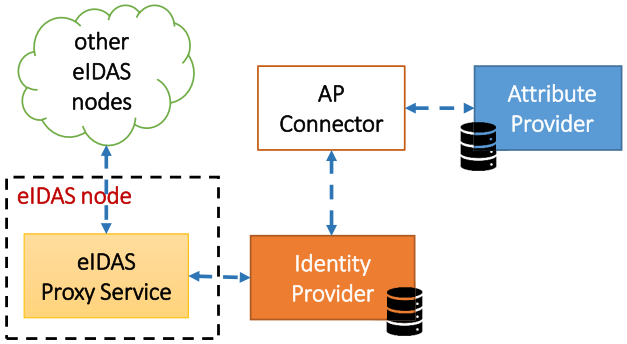

FIGURE 4. AP connector placed in between the identity provider and the attribute provider.

as depicted in Fig. 4. Note that the communication of the IdP with the national APs is MS specific, and it can be implemented in different ways. For example, the IdP and the APs could exploit a SAML 2.0 based protocol as well.

In the eID4U project, the first model has been adopted by Politecnico di Torino (Italy), Universidad Politécnica de Madrid (Spain) [64], Jozef Stefan Institute (Slovenia) [65], and Technical University of Graz (Austria). The second model was adopted by University of Lisbon (Portugal). A simplified AP Connector implementation has been described briefly in [18] and [19].

VI. AP CONNECTOR INTEGRATION WITH THE ITALIAN eIDAS NODE

In this section, we describe the integration of the Italian eIDAS infrastructure with the AP Connector and with the SPID identity system. To allow the Italian eIDAS node to communicate to the SPID IdPs, the FICEP project [66] has designed and implemented a dedicated component in the Specific part of the node, namely the IdP Proxy. This component allows the citizen to select the SPID IdP to authenticate with his credentials. Moreover, it converts the eIDAS messages to the SPID ones (and vice-versa).

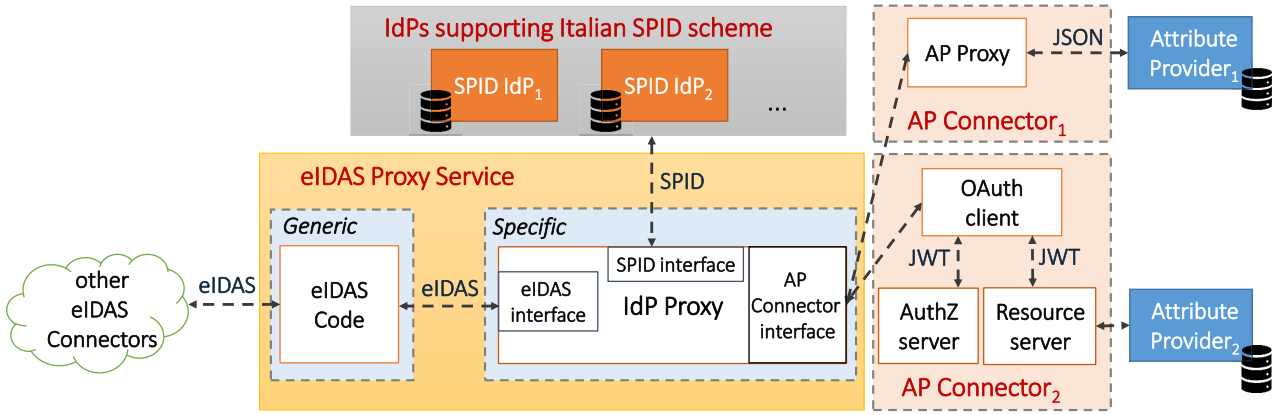

FIGURE 5. Integration of the (logical) AP connectors with the Italian eIDAS node.

First, we have extended the eIDAS code to support the attributes shown in Table 4. Then, we modified the IdP Proxy to get the additional attributes from a national AP, aggregate them with the ones valued by the SPID IdP, and convert them into the eIDAS format. The components of the Generic and Specific parts of the eIDAS Proxy Service involved in this process are shown in Fig. 6, while the workflows executed in the Generic and Specific parts of the eIDAS node are given respectively in Section VI-B and Section VI-C.

To identify the natural persons in the attribute retrieval process, we have used the eIDAS MDS attributes plus the (Italian) fiscal number assigned by the national Tax Agency. The fiscal number is single, unique and permanent, that is its value does not change in time. We have designed a new interface in the IdP Proxy, named AP Connector interface, as shown in Fig. 5.

In the first AP Connector implementation, named *AP-Proxy*, the eIDAS node communicates directly with a so-called AP Proxy module on a dedicated backed channel to obtain the attributes. In this implementation, the AP returns to the AP Proxy module more attributes than requested. Thus, the AP Proxy module filters the extra attributes. In the second AP Connector implementation, named *AP-OAuth2*, the AP and the eIDAS node supports the OAuth 2.0 protocol for better authorization of the data released and user consent management.

A. THE ITALIAN SPID SYSTEM IN BRIEF

To exploit the SPID system, the citizen must register with an authorized SPID IdP that has to identify him before releasing a SPID credential. Note that a citizen may register with different SPID IdPs, and may obtain several SPID credentials of different security levels. Once the citizen receives a SPID credential, he may access services provided by the SPID-enabled SPs. Since the support for SPID has become mandatory in the public services, many Italian entities such as municipalities, universities, the tax agency, hospitals, or other governmental agencies allow citizens' authentication through the SPID system.

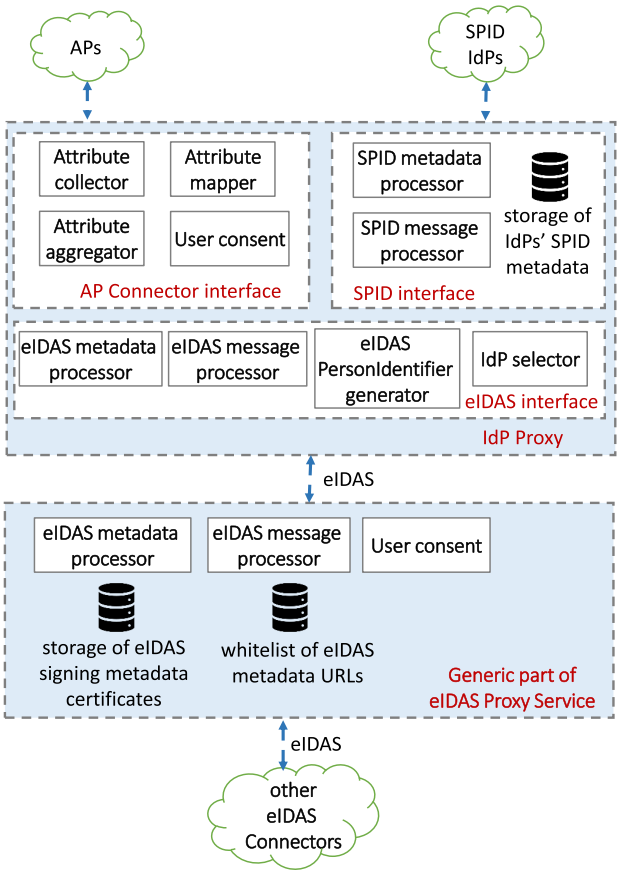

FIGURE 6. Internal view of the Italian eIDAS node with the components in the generic and the specific parts of the eIDAS proxy service.

SPID Protocol: Technically speaking, the SPID system is based on the SAML 2.0 standard as well, the operating modes are those provided by SAML v2 for the "Web Browser SSO" profile. When the citizen accesses a SPID-enabled service, the SP generates a digitally signed SPID authentication request (*SPID-Auth-Req*) and sends it to the SPID IdP, where the citizen authenticates with his SPID credential. On successful authentication, the SPID IdP sends back

TABLE 5. SPID attributes. Mandatory attributes for the user registration are marked with an asterisk.

Name	Description	Type	Example
spidCode	Identification code assigned by the SPID IdP, must be unique in SPID system. The format is: <cod_IdP><unique. number>, where <cod_IdP> is a 4-letter string uniquely assigned to the SPID IdP and the <unique. number> is an unique 10-characters long string, generated by the SPID IdP	xs:string	ABCD123456789A
familyName*	String composed of one or more non-empty substrings, separated by one single space	xs:string	Rossi
name*	String composed of one or more non-empty substrings, separated by one single space	xs:string	Marco
dateOfBirth*	Date in the format YYYY-MM-DD.	xs:date	1980-11-05
countyOfBirth*	Two-characters code of the Italian province of birth	xs:string	MI
placeOfBirth*	Four-characters code of the Italian municipality of birth	xs:string	A010
address*	String composed of one or more non-empty substrings, separated by one single space containing specific information for address (e.g., street/city square, house number, postal code, city and Italian province)	xs:string	Via Po 3 12042 Bra CN
gender*	Gender. Values accepted: F or M	xs:string	M
fiscalNumber*	Unique, single and permanent fiscal identification number, in the format proposed by Draft ETSI EN 319 412-1	xs:string	TINIT-RSSMRC80S05A010D
mobilePhone*	Phone number expressed as a numerical string with no spaces	xs:string	1234567890
email*	Email address in standard email address format	xs:string	marco.rossi@example.com
idCard*	String composed of four fields: idType, idNumber, idIssuer, idIssueDate, idExpirationDate	xs:string	Carta d'identità WX12345XZ Comune di Roma 2015-10-23 2025-11-05
expirationDate	Expiration date of the SPID identity in the format YYYY-MM-DD	xs:date	2022-05-24
digitalAddress*	Certified email address (PEC, in Italy)	xs:string	marco.rossi@pec.example.it

a digitally signed SPID authentication response (*SPID-Auth-Res*) message to the SP, which provides or denies access to the service based on the response received.

SPID Metadata: The certificate(s) used to verify the signatures of the SPID messages are distributed via specific SAML metadata. In practice, the SPID IdPs and SPs share the (SPID) SAML metadata through a SPID Registry [68] maintained by AgID (Agenzia per l'Italia digitale) [67]. This entity performs verifications so that only the accredited IdPs and the verified SPs can connect to the SPID system. Moreover, it defines the SPID messages format.

SPID Attributes: At authentication time, the IdP transfers the so-called SPID attributes to the SPs. In Table 5, we indicate the ones defined for the natural persons [69]. The attributes are grouped in SPID *attribute sets*. Each SP defines in its own SAML metadata one or more attribute sets, and each set is identified by a numeric index. When the SP creates the *SPID-Auth-Req* it inserts in the request the index number of a SPID attribute set. In this way, the SPID IdP determines the requested attributes based on the attribute set index contained in the SAML metadata of that SP. For the interaction between the IdP Proxy acting as a SPID SP in the communication with the SPID IdPs, we have defined a new attribute set and the corresponding index. This set contains the minimum trustworthy information about a person (name, surname, date of birth) plus the fiscal number.

B. THE GENERIC PART OF THE ITALIAN eIDAS NODE

The Italian eIDAS node processes an *eIDAS-Auth-Req* received from a corresponding eIDAS Connector by executing the steps shown in Fig. 7 with the node components shown in Fig. 6. Since the request is digitally signed, the node must get first the $Cert_{eIDAS\_message\_sign}$ certificate of the eIDAS Connector. This certificate is part of the Connector's eIDAS metadata, which is downloaded and validated with the eIDAS metadata processing module as detailed in Fig. 8. If the eIDAS metadata has been successfully verified, the eIDAS message processor module prepares next the *eIDAS-Auth-Req*. In practice, the node creates a new *eIDAS-Auth-Req*, which will be sent to the IdP Proxy. To this aim, the node retrieves and validates the IdP Proxy's eIDAS metadata, by executing the steps in Fig. 9. Then, it checks whether the requested attributes are present in the IdP Proxy's metadata. The new *eIDAS-Auth-Req*, digitally signed with the private key corresponding to the $Cert_{eIDAS\_message\_sign}$ of the Generic part, is sent to the IdP Proxy which processes it as described in Section VI-C.

Once an *eIDAS-Auth-Res* is received back from the IdP Proxy, the node retrieves and validates IdP Proxy metadata (as shown in Fig. 8). Next, it decrypts the eIDAS attribute values with the corresponding encryption certificate, and it generates a user consent form for the decrypted attributes. Finally, it retrieves and validates the Connector's eIDAS metadata, then it creates a new *eIDAS-Auth-Res* holding the attributes encrypted with the $Cert_{eIDAS\_attributes\_encrypt}$ certificate of the Connector. In the last step, the digitally signed *eIDAS-Auth-Res* is sent back to the eIDAS Connector.

C. THE SPECIFIC PART OF THE ITALIAN eIDAS NODE (IdP PROXY)

The IdP Proxy communicates with the SPID IdPs and retrieves additional attributes by running the steps shown in Fig. 10 and Fig. 11 respectively. First, the *eIDAS-Auth-Req* received from the Generic part is validated by using the corresponding eIDAS metadata, which is downloaded and processed on the fly, as shown in Fig. 8. Next, the IdP

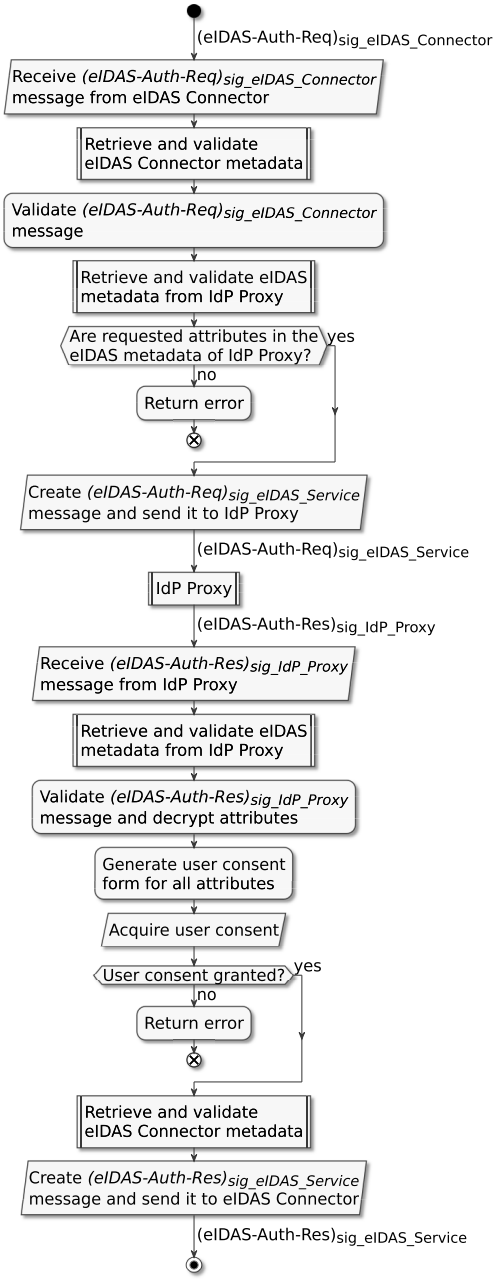

FIGURE 7. Steps performed in the generic part of eIDAS Proxy service. Notation: $(A)_{sig\_B}$ means message A is digitally signed by entity B.

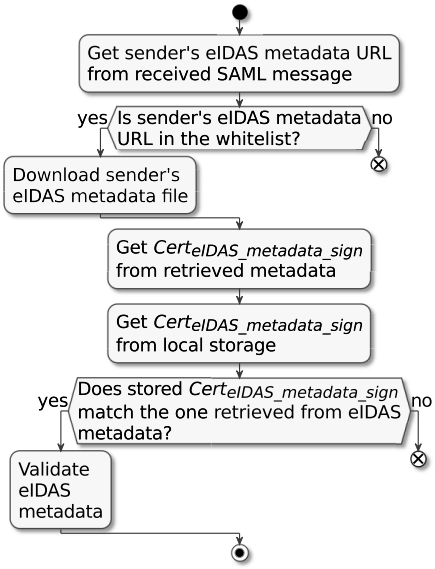

FIGURE 8. Validation of the eIDAS metadata upon receiving an eIDAS message from a sender.

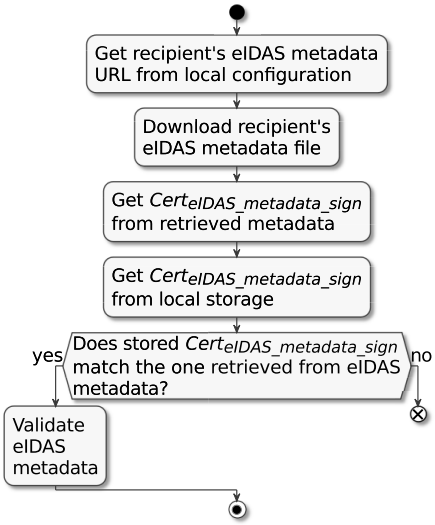

FIGURE 9. Validation of the eIDAS metadata upon sending a message to a recipient.

Proxy converts the attributes from the eIDAS format into the SPID one. If the request contains attributes that cannot be valued by the SPID IdP, then the IdP Proxy adds the SPID `fiscalNumber` to the list of requested attributes to identify the citizen in the attribute retrieval phase. Subsequently, the IdP Proxy downloads the SPID IdP metadata and checks whether the requested attributes are contained therein. Then, it selects the SPID attribute set index, it creates a signed *SPID-Auth-Req*, and sends it to the SPID IdP.

The received *SPID-Auth-Res* is processed as shown in Fig. 11. The message is first validated by exploiting the corresponding SPID IdP metadata. If there are attributes that have not been valued by the SPID IdP, they are retrieved next from the AP either with the AP Proxy (described in Section VII-B), or the AP-OAuth2 approach (described in Section VII-C). The returned attributes are converted into the eIDAS format, and are encrypted by exploiting the corresponding $Cert_{eIDAS\_attributes\_encrypt}$ of the Generic part. Finally, the IdP Proxy creates a new digitally signed *eIDAS-Auth-Res*, which is sent to the Generic part of the eIDAS node.

VII. AP CONNECTOR IMPLEMENTATION DETAILS

This section provide details on the proposed AP Connector implementations, as well as the modifications of the eIDAS code in the Generic and Specific parts. We have used the

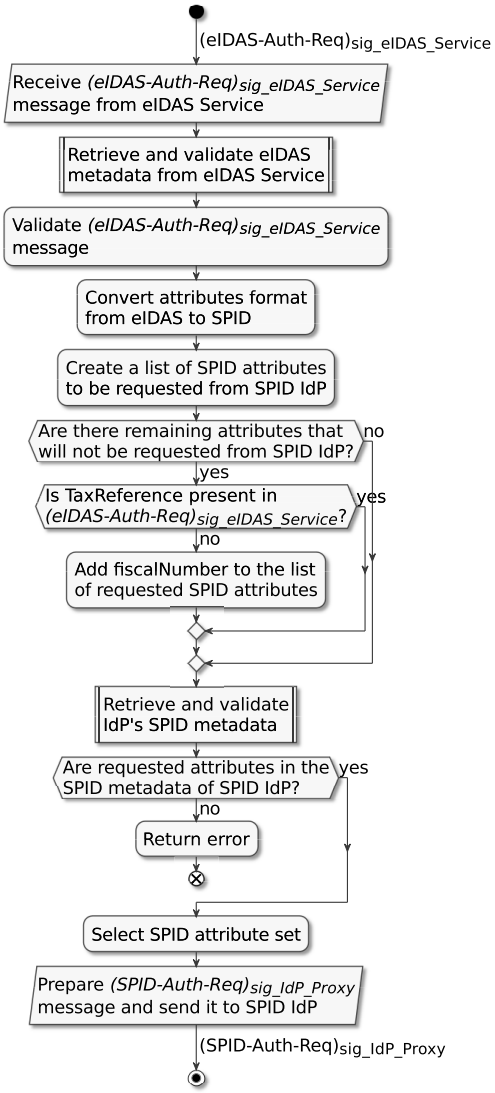

FIGURE 10. Steps performed in the specific part of eIDAS Proxy service (namely in the IdP Proxy) to handle the eIDAS authentication request. Notation: $(A)_{sig\_B}$ means message A is digitally signed by entity B.

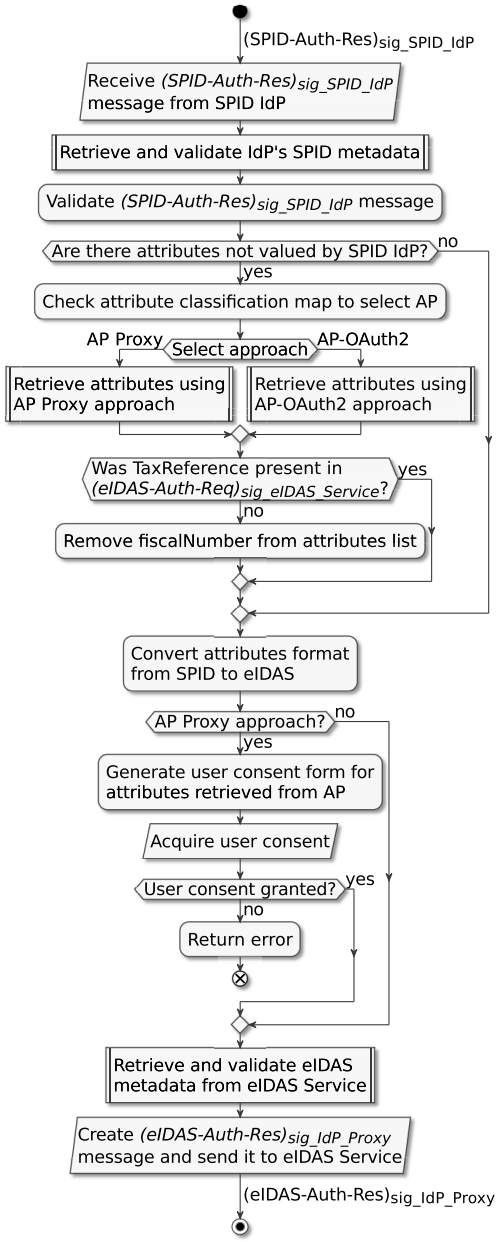

FIGURE 11. Steps performed in the specific part of eIDAS Proxy service (namely in the IdP Proxy) to handle the *SPID-Auth-Res* and the attribute retrieval by using either the AP Proxy or the AP-OAuth2 approach. Notation: $(A)_{sig\_B}$ means message A is digitally signed by entity B.

eIDAS code v1.4.4 [70], [71], which uses Apache Tomcat 8 and Apache Struts 2 framework [72]. We have deployed the implemented AP Connector components on the eIDAS node and on dedicated machines at our site. Note that both the SPID IdP and the AP backend (i.e., the university database) are out of our control.

A. ENABLING NEW ATTRIBUTES ON THE eIDAS NODE

The Generic part of the eIDAS node allows support for new attributes of common types. We filled in the empty configuration file named *saml-engine-additional-attributes.xml* with the details of the attributes defined in [18] and reported in Table 4. For example, we defined the `TaxReference` attribute for the natural person in the following way:

```
<entry key="14.NameUri">
  http://eidas.europa.eu/attributes/
    naturalperson/TaxReference
</entry>
<entry key="14.FriendlyName">
  TaxReference
</entry>
<entry key="14.PersonType">
  NaturalPerson
</entry>
<entry key="14.Required">false</entry>
<entry key="14.XmlType.NamespaceUri">
  http://eidas.europa.eu/attributes/
    naturalperson
</entry>
```

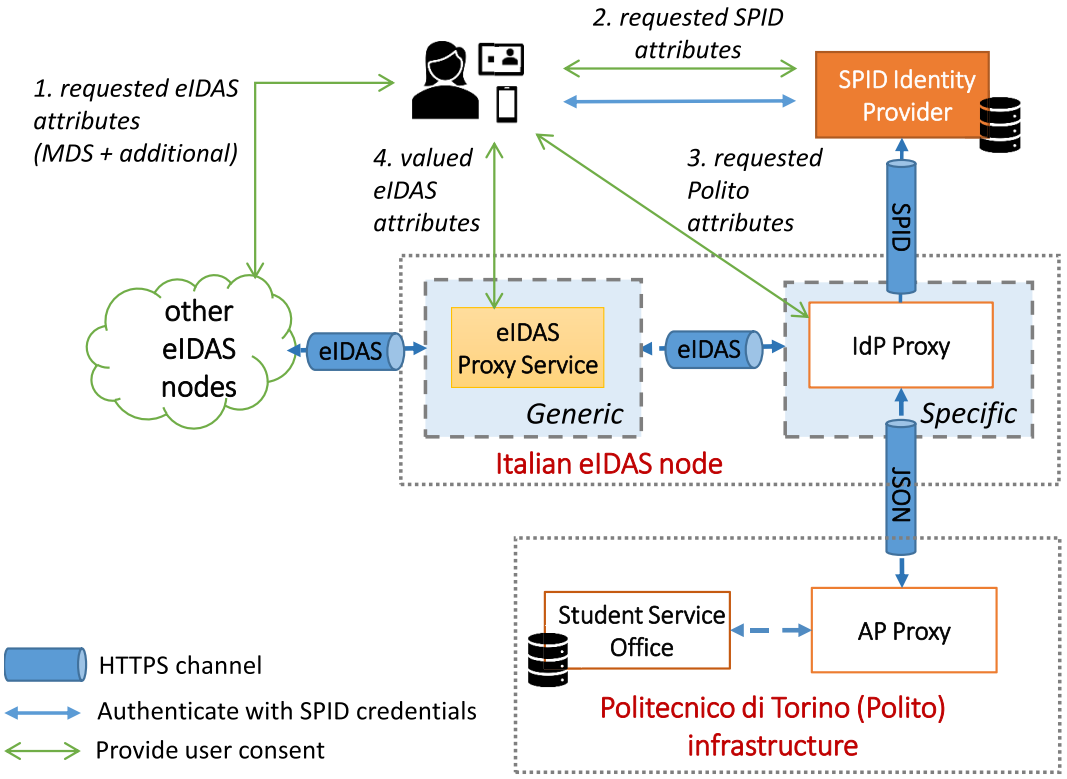

FIGURE 12. Components involved in the AP Proxy approach implementation. The IdP Proxy, running in the specific part of the eIDAS Node, communicates over the SPID interface with the SPID IdPs. The AP Proxy exchanges attributes in JSON format with the IdP Proxy over a directly established HTTPS channel (no user interaction). Attributes returned by the AP are filtered and converted into an eIDAS compatible format on the AP Proxy. Finally, the IdP Proxy aggregates the SPID attributes with the ones valued by the AP. Finally, it generates an *eIDAS-Auth-Res* returned to the Generic part of the Italian eIDAS node.

```
<entry key="14.XmlType.LocalPart">
  TaxReferenceType
</entry>
<entry key="14.XmlType.NamespacePrefix">
  eidas-natural
</entry>
<entry key="14.AttributeValueMarshaller">
  eu.eidas.auth.commons.attribute.impl.
    LiteralStringAttributeValueMarshaller
</entry>
```

We modified the IdP Proxy classes to support attribute retrieval from the AP, the attribute aggregation, and the user consent in the AP Proxy approach.

The IdP Proxy is composed of three different classes: a) the **EIDASController** class implements the eIDAS interface to communicate with the eIDAS Proxy-Service; b) the **SPIDController** class implements the SPID interface to communicate with the SPID IdPs, while the **IdpProxyService** class converts the messages from the eIDAS protocol to the SPID protocol and viceversa. To support the user consent on the IdP Proxy, we added new logic in the **IdpProxyService** class.

B. AP PROXY APPROACH

For this approach, we designed the solution shown in Fig. 12. Both the sequence diagram and the implementation details are explained further below.

We implemented the AP Proxy application as a RESTful Web service [73] by using the Flask web application framework [74] for Python. On the eIDAS node, we have added a new class in the AP Connector interface of the IdP Proxy, called **APProxyRequestData**. The **SPIDController** class invokes the above class by passing it the fiscal number of the person and the list of requested attributes, as shown in Fig. 13. Next, the **APProxyRequestData** establishes a mutually authenticated TLS connection with the AP Proxy application. Over the TLS protected channel, the application sends an HTTP GET request holding the fiscal number and the additional requested attributes. The AP Proxy application reads the fiscal number and sends it through an HTTP GET request to a dedicated web service running at the Student Service Office backend. The same method receives back the attributes contained in a JSON response [17]. For example, the attributes

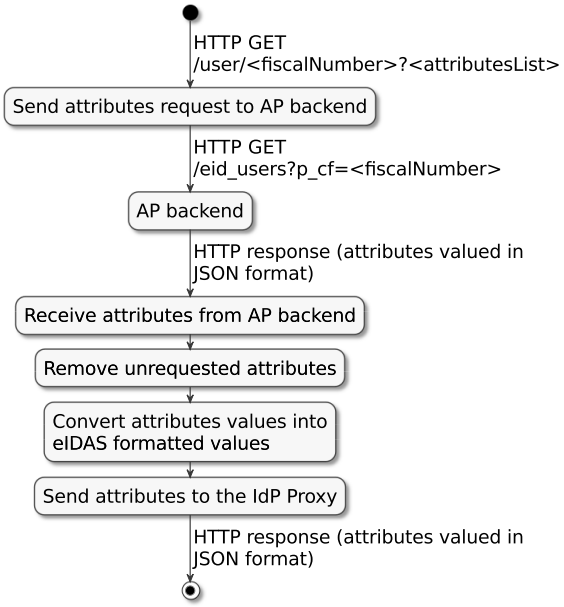

FIGURE 13. AP Proxy flowchart.

(in JSON format) returned to the AP Proxy application are shown below:

```
{
 "Citizenship": "IT",
 "CountryOfBirth": "IT",
 "CurrentAddress": "QklUVEk=",
 "CurrentDegree": "",
 "CurrentFamilyName": "ROSSI",
 "CurrentGivenName": "MARCO",
 "DateOfBirth": "1994-03-29",
 "Email": "s111122@studenti.polito.it",
 "Gender": "Male",
 "HomeInstitutionAddress": "Q2..JUxZ\f\n",
 "HomeInstitutionCountry": "IT",
 "HomeInstitutionIdentifier": "POLITO",
 "HomeInstitutionName":
"Politecnico di Torino",
 "IdIssuer": "MCTC-NU",
 "IdNumber": "NU5341568Z",
 "MaritalState": "N/A",
 "Nationality": "IT",
 "PersonIdentifier": "176311",
 "Phone": "3465678312345",
 "PlaceOfBirth": "Milano",
 "TemporaryAddress": "QklUVEk="
}
```

The AP Proxy application discards the not-requested attributes, while the requested ones are changed into an eIDAS compatible format. For example, the attributes containing an address (e.g., CurrentAddress) are inserted in a `<eidas-natural:FullCvaddress>` tag. The attributes (in JSON format) are sent back to the IdP Proxy. The **APProxyRequestData** class parses the received JSON answer, and returns the valued attributes to the **SPIDController** class.

To establish mutually authenticated TLS connections between IdP Proxy and AP Proxy, we have obtained TLS certificates from Let's Encrypt Certification Authority (CA), and we configured them into the deployed components. The NGINX reverse proxy [75] placed in front of the AP Proxy works as a TLS termination proxy [76]. We added the reverse proxy's TLS certificate into the IdP Proxy's dedicated certificate keystore and the IdP Proxy's TLS certificate into the reverse proxy's certificate keystore.

The user consent is asked in four different steps, as shown in Fig. 12. The eIDAS Proxy Service generates user consent pages for the eIDAS MDS mandatory attributes (shown in Fig. 14 (a)), and for the optional eIDAS MDS attributes and the additional ones (shown in Fig. 14 (b)). The SPID IdP creates a user consent page for the requested SPID attributes (Fig. 14 (c)), while the IdP Proxy creates user consent pages for the attributes requested from the AP (Fig. 14(d)). Finally, the eIDAS Proxy Service asks the user consent for the valued attributes (Fig. 14 (e)) before sending them to the counterpart eIDAS node.

C. AP-OAuth2 APPROACH

To support this approach, we designed and implemented the architecture shown in Fig. 15, which exploits the Authorization (AuthZ) Code Grant flow of the OAuth 2.0 protocol.

1) OAuth2 AuthZ CODE GRANT FLOW

The OAuth2 AuthZ Code Grant flow, described in RFC-6749 [16], is the most used grant type of OAuth2 protocol. The flow considers four roles:

- The Resource owner (the user) is the entity authorizing access to protected resources via a user-agent (Web Browser).
- The (OAuth) client is the entity requesting access to protected resources.
- the authorization (AuthZ) server is the entity handling authorization to the protected resources and the user consent. It issues an `authorization code` (AuthZ code), which is consumed by the OAuth client in exchange of an `access token`. The AuthZ server sends the access token to the client after checking the Resource owner's authentication and (granted) authorization.
- the Resource server is the entity hosting the protected resources and allowing access to them after validating the access token.

2) AP-OAuth2 SEQUENCE DIAGRAM

In the testbed, we have deployed the (OAuth) client in the Specific part of the eIDAS node, while the AuthZ server and Resource server components run at the university premises. The sequence of steps performed in this approach are shown in Fig. 16, and are described below.

First, the IdP Proxy sends the (list of) requested attributes and the citizen identification information containing the SPID fiscal number to the OAuth client (step 1). Then the client

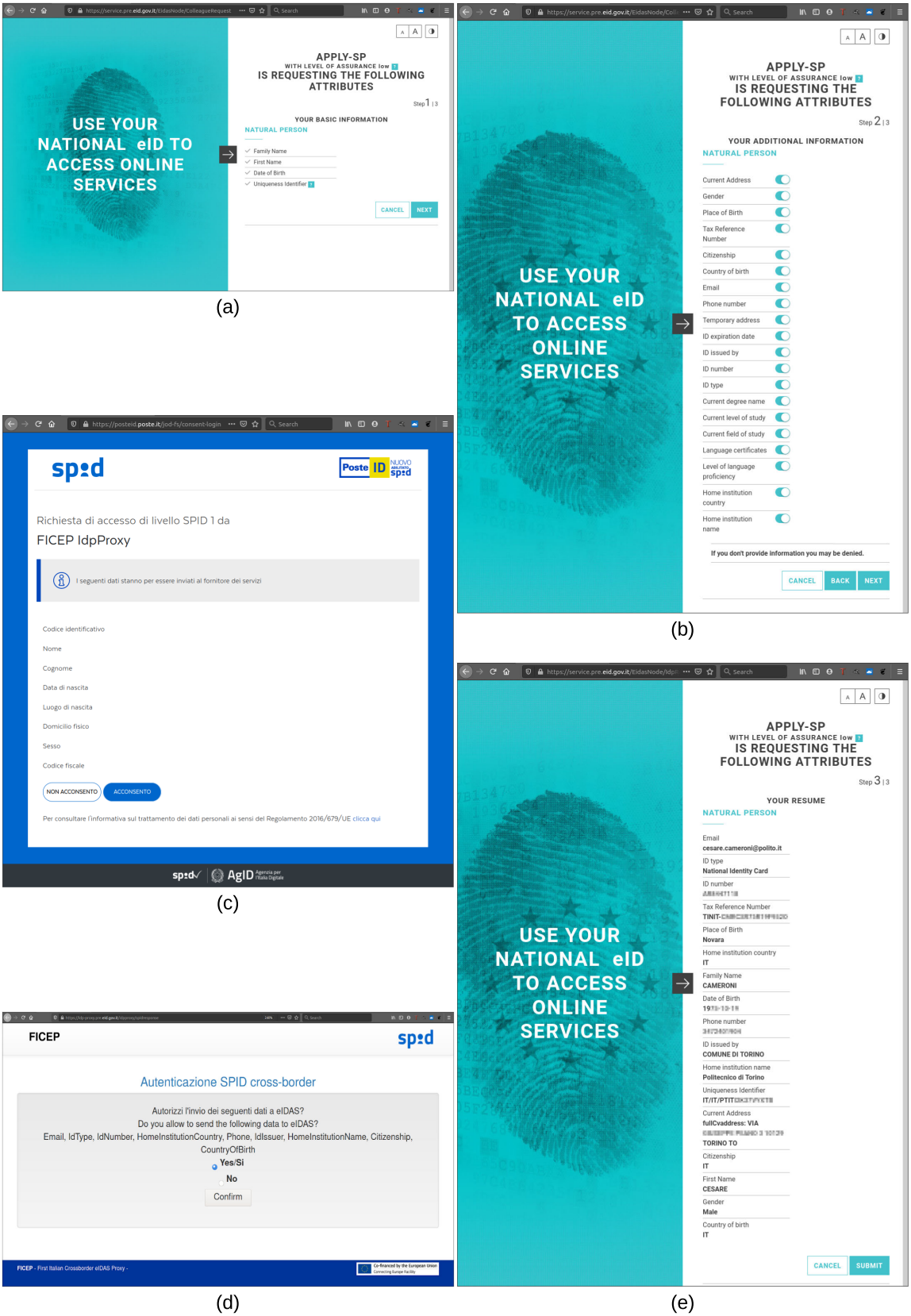

FIGURE 14. User consent screenshots: a) consent for mandatory eIDAS MDS attributes on eIDAS Proxy service; b) consent for the optional eIDAS MDS attributes and additional (defined) attributes on eIDAS Proxy service; c) consent for the requested SPID attributes on a SPID IdP; d) consent for additional attributes requested from an attribute provider on IdP Proxy; e) consent for valued attributes on eIDAS Proxy service.

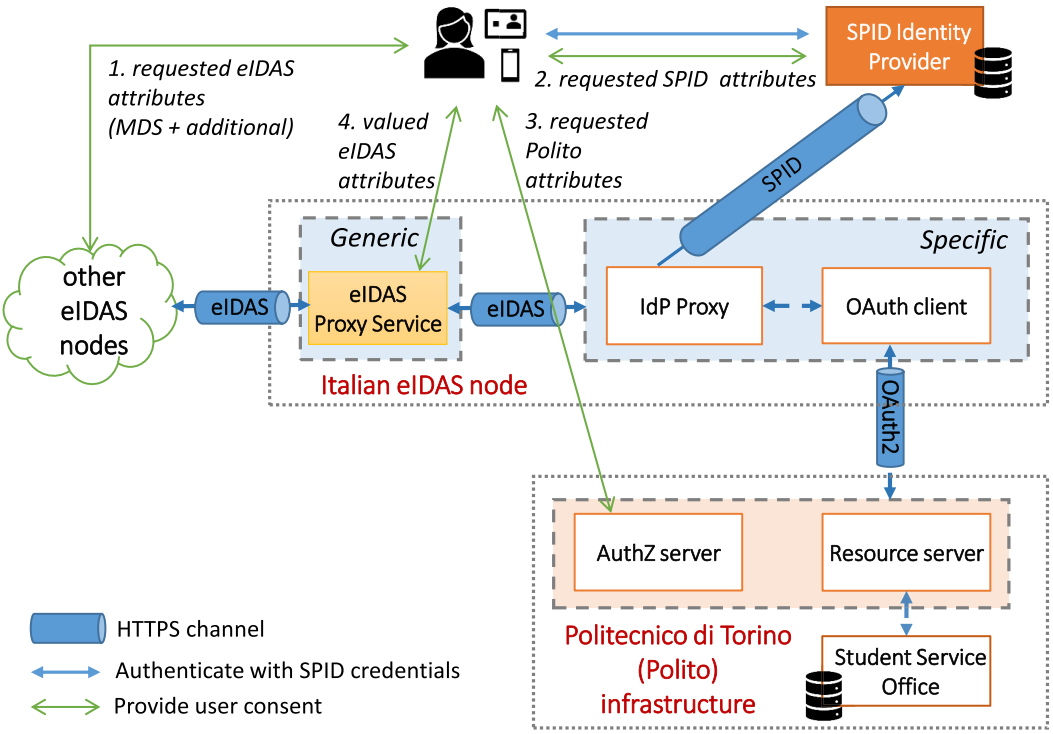

FIGURE 15. Components involved in the AP connector implementation exploiting OAuth 2.0 protocol.

starts the OAuth 2.0 Authorization Code Grant flow by redirecting the user browser to the AuthZ server (step 2), which authorizes access to the user's attributes. Note that the `scope` parameter of the authorization request in step 2 holds the requested attributes. In steps 3-4, the AuthZ server interacts with the OAuth client to retrieve the citizen identification data (stored in a digitally signed identity token `id_token`) because it relies on the authentication previously performed with the SPID IdP. After validating the received identity token (step 5), the AuthZ server interacts asks the user to provide the consent on the requested attributes (steps 6-8), e.g., `IdNumber` or `HomeInstitutionName`. To this aim, it generates an HTML web form with individual checkboxes for each requested attribute. Then the AuthZ server generates the authorization code, which is sent back to the OAuth client via user browser redirection (step 9). Based on the authorization code, the OAuth client sends an Access Token request to the AuthZ server (step 10), along with the `client_id` and `client_secret` parameters used for OAuth client authentication. The Access Token response sent back by the AuthZ server (in step 11) contains the access token, which is a digitally signed JSON Web Token (JWT). The access token consists of: a) the citizen identification data (in the field "sub" of the token), and b) the attributes that have been authorized (in the field "scope" of the token).

In the steps 12-14, the OAuth client interacts with the Resource server, which retrieves the attributes from the AP backend. More specifically, the OAuth client sends the access token to the Resource server (step 12). After validating the access token, the Resource server gets the attributes from the AP backend, it filters the unrequested attributes, and converts them in JSON format. Finally, it sends the attributes back to the OAuth client (step 14), which forwards them to the IdP Proxy (step 15).

3) AP-OAuth2 IMPLEMENTATION DETAILS

To support this approach, we modified first the IdP Proxy, in particular the `processResponse` method in the **SPIDController** class. This method receives the *SPID-Auth-Res* from the SPID IdP, it gets the citizen's identification data (i.e., SPID fiscal number) and calls the OAuth client, which starts the attribute retrieval via the OAuth2 protocol. On the way back, the OAuth client sends the returned attributes to the **SPIDController** class. This method performs the attributes conversion into eIDAS format and the aggregation with the previously retrieved SPID attributes. In the last step, the IdP Proxy prepares an *eIDAS-Auth-Res* message, and sends it to the Generic part of the eIDAS Proxy Service.

Moreover, we implemented the OAuth client, AuthZ server, and Resource server components, as described below. In brief, we have used Maven [77] for project management and dependencies, Jakarta EE [78] with MicroProfile [79], and Open Liberty [80] as server runtime environment. For JWT token management, we used the Nimbus JOSE (Javascript Object Signing and Encryption) and JWT library [81].

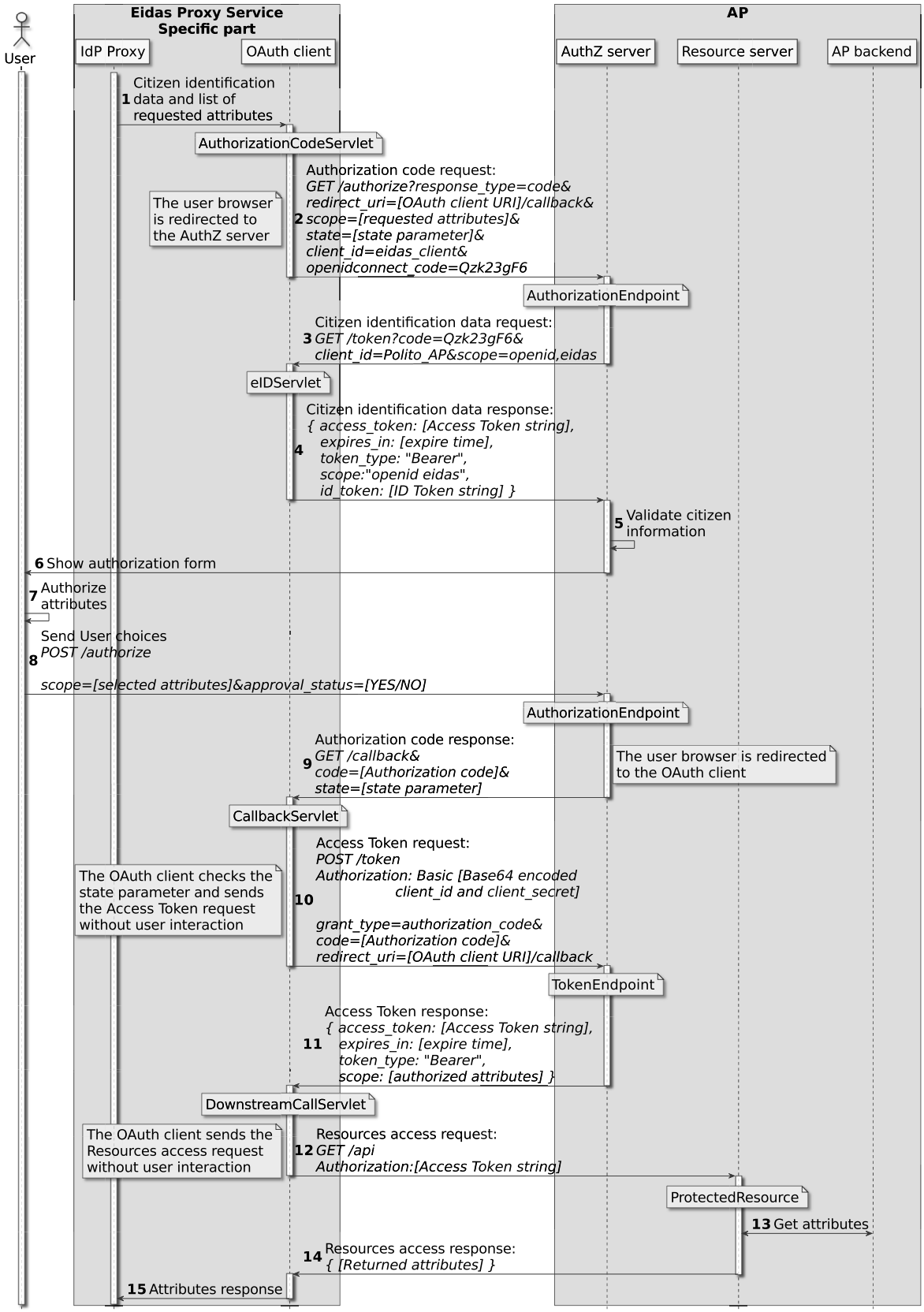

FIGURE 16. AP-OAuth2 approach flowchart.

a: OAuth CLIENT IMPLEMENTATION

The OAuth client is implemented through four servlets (shown in Fig. 16) that exploit the following configuration parameters stored in a dedicated MicroProfile Config file: (a) *client_id* is the OAuth client identifier, such as *eidas_client*; (b) *client_secret* is the secret shared between the OAuth client and the AuthZ server; (c) *redirect_uri* is the endpoint where to receive the authorization code from AuthZ server; (d) *authorization_uri* is the AuthZ server endpoint where the request for the authorization code is sent; (e) *token_uri* is the AuthZ server endpoint where the request for the access token is sent; (f) *resourceserver_uri* is the Resource server endpoint where the access request for the additional attributes is sent.

The first servlet, named **AuthorizationCodeServlet**, is invoked by the IdP Proxy on the `/authorization` endpoint. The servlet receives the set of requested attributes and the citizen identification data passed in as HTTP parameters. Then, it creates and stores the `state` parameter, it reads the configuration information from the MicroProfile Config file, and saves the required attributes in the `scope` parameter, e.g.

```
scope= IdNumber,
  HomeInstitutionName,
  HomeInstitutionIdentifier
```

Then, it builds the URI attaching all the above variables and it redirects the user to the configured AuthZ server endpoint.

The second servlet, named **eIDServlet**, is invoked by the AuthZ server on the `/token` endpoint. The servlet receives the citizen identification request and returns an identity token in form of a signed JSON Web Token (JWT) containing the citizen identification data, including his fiscal number. The third servlet, named **CallbackServlet**, waits for the authorization code on the `/callback` endpoint. It checks the received `state` parameter against the saved one, and it uses the received authorization code to request an access token at the `/token` AuthZ server endpoint. The fourth servlet, named **DownstreamCallServlet**, waits for the access token on the `/downstream` endpoint. The OAuth client sends the access token to the `/api` endpoint of the Resource server to get the requested attributes from the AP backend.

b: AuthZ SERVER IMPLEMENTATION

It is composed of two classes shown in Fig. 16, namely the **AuthorizationEndpoint** and the **TokenEndpoint**. The **AuthorizationEndpoint** waits for authorization code request sent via an HTTP GET to the `/authorize` endpoint. It validates the OAuth client by checking the client_id and client_secret, the redirect URI, the requested scope (holding the attribute names) and saves all these parameters in the session. Subsequently, it request the citizen identification data from the OAuth client. In practice, it requests claims about user's identity in a format defined by OpenID Connect [82]. The response in step 4 is contained in an ID token [83], which is an encoded and signed JWT, sent along with an access token. After validating the ID token (in step 6), it redirects the user browser to the web page where the user provides the consent for the requested attributes.

The `doPost` servlet of the **AuthorizationEndpoint** class waits for an HTTP POST on the `/authorize` endpoint. Upon receiving the user consent from the authorization web page, it uses the previously saved parameters to populate the authorization code. Next, it redirects the user browser back to the OAuth client (on the redirect_uri) by adding to the query parameters the authorization code alongside the `state` parameter.

The OAuth client checks the state parameter and sends the access token request to the AuthZ server, without browser intervention. The **TokenEndpoint** class waiting for an HTTP POST on the `/token` endpoint authenticates the OAuth client by checking the client_id and client_secret. Then, it creates the access token in JWT format [84], and sends it to the AuthZ server. An example of the response sent in step 11 is shown below:

```
{
  "access_token": "84wfeKwT0zdeinfYfw\ldots",
  "token_type": "Bearer",
  "expires_in": 3600,
  "scope": "IdNumber,
    HomeInstitutionName,
    HomeInstitutionIdentifier"
}
```

c: RESOURCE SERVER IMPLEMENTATION

This server exposes the `/api` to the OAuth client to access user attributes by consuming the access token. The **ProtectedResource** class checks the authorization by validating the received access token. Next, it interacts with the AP to retrieve the attributes, it removes the attributes that have not been requested, and converts the attribute values into an eIDAS compatible format. Finally, the attributes (in JSON format) are returned to the OAuth client. An example of the returned attribute response is:

```
{
  "IdNumber": "123241234973",
  "HomeInstitutionName":
    "Politecnico di Torino",
  "HomeInstitutionIdentifier": "IT"
}
```

In this implementation, we used JAX RS [85] to access web resources and MicroProfile JWT [86] to validate and the map the scopes in Jakarta roles.

A comparison of the two approaches is given in Table 6.

VIII. VALIDATION OF RESULTS

To validate the transfer of additional attributes, we connected the modified Italian eIDAS node (supporting the new attributes) to the corresponding extended nodes in Portugal, Spain, and Slovenia, as shown in Fig. 17. For this scope, AgID has deployed our modified versions of the eIDAS node and of the IdP Proxy in the Italian eIDAS

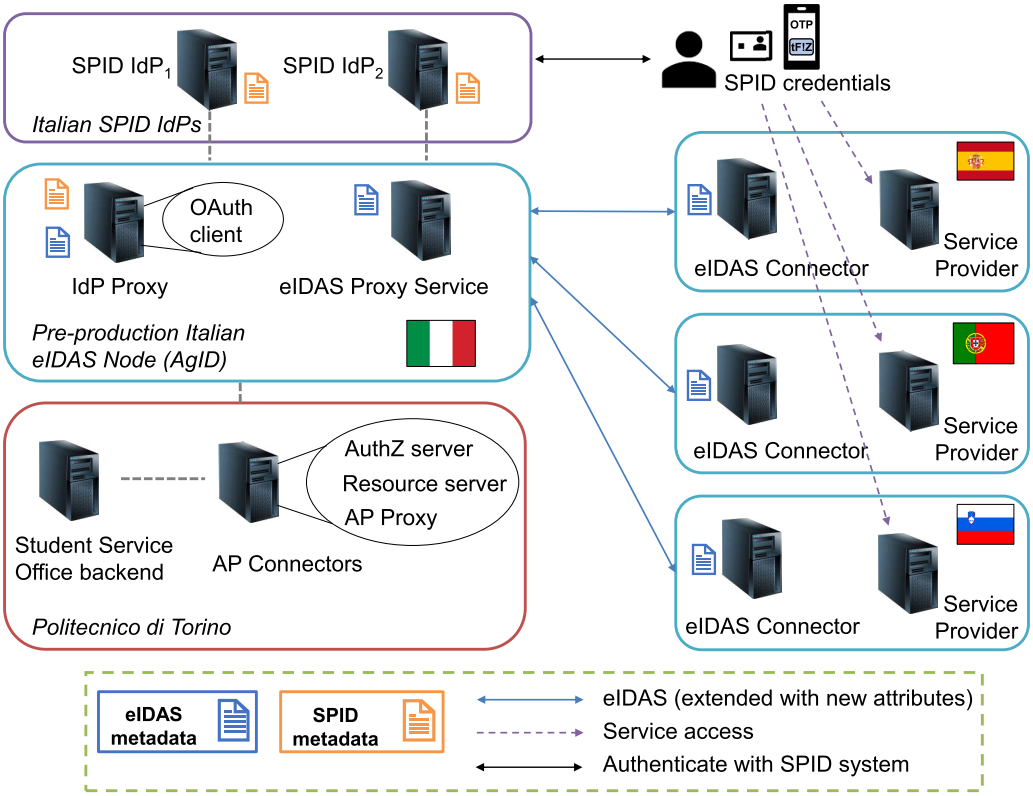

FIGURE 17. Experimental testbed.

TABLE 6. Comparison of the two proposed solutions in terms of attribute filtering, attribute aggregation with the attributes valued by the SPID IdP, the conversion from Polito format into eIDAS format, and the user consent for the attributes returned from the AP. Notation: X(Y), where X is the entity performing the operation and Y is the module in charge of the operation.

Operation	AP-Proxy approach	AP-OAuth2 approach
Attributes filtering	Polito (AP Proxy)	Polito (Resource server)
Attributes aggregation	eIDAS node (IdP Proxy)	eIDAS node (IdP Proxy)
Attributes conversion	Polito (AP Proxy)	Polito (Resource server)
User consent	eIDAS node (IdP Proxy)	Polito (AuthZ server)

pre-production environment. The partner universities, Universidad Politécnica de Madrid (Spain), University of Lisbon (Portugal), and Jozef Stefan Institute (Slovenia), have designed and implemented dedicated Erasmus eIDAS-enabled student enrollment services for foreign students.

Each university has specific administrative procedures and technical workflows to manage the Erasmus applications. Anyhow, the collected data containing both personal and academic status is part of a so-called ''Erasmus+ application process creation'', which typically generates a new account for the foreign student at the visiting university. The International Relations Office and the Academic Office personnel must validate the Erasmus+ application to complete the registration. The foreign universities have deployed the three services on dedicated endpoints indicated in Table 7. The attributes have been classified as mandatory (M) or optional (O) in the services. We distinguish the mandatory attributes for the services from the mandatory ones for eIDAS. The first ones are required to provide the service, while the eIDAS mandatory attributes are always transferred (upon authentication through the eIDAS network).

Experimental Testbed Setup: We deployed the AP Proxy, AuthZ server, and Resource server applications on a dedicated machine, as shown in Fig. 17. To support the new attributes, we modified the eIDAS metadata as described in Section VII-A. We have updated the SPID metadata with a SPID attribute data set containing the (Italian) fiscal number. We have configured the OAuth client, the AuthZ server, and the Resource server with the parameters required for OAuth client authentication and processing of the identity and access tokens.

Validation Sessions: We have involved 30 students registered at Politecnico di Torino (Polito) to perform the Erasmus student enrollment at the above universities. Our students have authenticated through eIDAS network with their SPID credentials, and their academic attributes have been retrieved from the university backend. Each test session lasted about thirty minutes and ended by filling out a survey. In the tests,

TABLE 7. Attributes transferred through the eIDAS network from the SPID IdP and Polito to foreign services. PT is the Portuguese service for Erasmus application (https://fenix-qua.igot.ulisboa.pt/eidas). ES is the Spanish service for Erasmus application (https://erasmus-eid4u.dit.upm.es/erasmus). SI is the Slovenian service for Eramus application (https://erazem.e5.ijs.si/eidas.php). Attributes marked with an asterix (*) are valued based on the ID documents and data provided by the student upon registration at our university.

eIDAS Attributes	Source	PT	ES	SI
PersonIdentifier	SPID IdP + eIDAS node	✓	✓	✓
FamilyName	SPID IdP	✓	✓	✓
FirstName	SPID IdP	✓	✓	✓
DateOfBirth	SPID IdP	✓	✓	✓
PlaceOfBirth	SPID IdP	✓	✓	✓
CurrentAddress	SPID IdP	✓	✗(2)	✓
Gender	SPID IdP	✓	✓	✓
TaxReference	SPID IdP	✓	–	–
IdType	Polito*	✓	–	✗(1)
IdNumber	Polito*	✓	–	✓
IdIssuer	Polito*	✓	–	✓
IdExpiryDate	Not provided			
EhicId	Not provided			
Nationality	Polito*	✓	–	✓
Citizenship	Polito*	✗(1)	–	✓
MaritalState	Not provided			
CountryOfBirth	Polito*	✓	–	✓
CurrentPhoto	Not provided			
TemporaryAddress	Not provided			
Email	Polito*	✓	–	✓
Phone	Polito*	✗(1)	✓	✓
HomeInstitutionName	Polito	✓	✓	✓
HomeInstitutionIdentifier	Polito	✓	–	✓
HomeInstitutionCountry	Polito	✗(1)	–	✓
HomeInstitutionAddress	Polito	–	✗(2)	✗(1)
CurrentLevelOfStudy	Not provided			
FieldOfStudy	Not provided			
CurrentDegree	Not provided			
Degree	Not provided			
DegreeAwardingInstitution	Not provided			
GraduationYear	Not provided			
DegreeCountry	Not provided			
LanguageProficiency	Not provided			
LanguageCertificates	Not provided			

(1) Attribute sent but not shown on the service.
(2) Address attributes validation failed on Spanish eIDAS node because of format change from eIDAS 2.2 onward.

the students have used their personal computers to access the services at the endpoints given in Table 7, by exploiting common browsers like Mozilla Firefox, Google Chrome, Microsoft Edge, or Apple Safari. Moreover, they did not have to apply any particular configuration. The SPID credentials used have been issued by different SPID IdPs, like InfoCert (4,35%), Aruba (8,7%), TIM (4,35%), Sielte (17,39%), or Poste (65,22%).

Attribute Transfer Results: Table 7 shows the attributes provided to the foreign services, both the ones valued by the SPID IdP, as well as the ones assessed by Polito. Some attributes have not been shown on the service final page because they haven't been converted correctly either on the eIDAS node or at the SP. The attributes marked with "-" were not requested by the service, while the ones "Not provided" were optional and were not valued by the AP. The eIDAS node valued the eIDAS PersonIdentifier based on the spidCode returned by the SPID IdP, along with the corresponding nationality codes.

User Feedbacks: In response to the question *"I think that including my academic profile provided by my national eID when accessing University e-services will improve the usability and the quality of those services"*, the majority (90,5%) of participants responded positively. However, a small number of participants have responded "No" (4,75%) or "I'm not sure" (4,75%).

In Response to the Question: I've used my eID in the Erasmus registration service and I think that the possibility of using my citizen eID extended with my academic profile facilitates the registration and improves the user experience, most of the participants responded positively (90,5%). Nevertheless, the remaining participants expressed doubts by responding "I'm not sure" (9,5%).

A participant mentioned that "the information-gathering process might be more straightforward", because the requested attributes were "not active" by default, and because the user consent pages occurred in several parts of the flow. Due to the usability and privacy concerns on the collected data, some participants responded negatively to the question "I would like the inclusion of this initiative in the academic service of other European universities".

Technical Team Feedbacks: We collected feedbacks from the Student Service Office (SSO) about the eIDAS-enabled application deployed on a dedicated endpoint (https://apply-eid4u.polito.it/SP/) to register students from Spain, Portugal, Slovenia, and Austria. The technical team considered it very useful that the identification and academic data about prospective students may be obtained in a trustworthy manner, avoiding thus performing several manual checks on the enrollment applications they receive each year.

IX. CONCLUSION AND FUTURE WORK

Citizen authentication and identification with a high assurance level are increasingly important in different types of services, either short-time or long-term ones. Although several steps have been accomplished in Europe to allow citizens' access to cross-border services with their eID(s), we assist nowadays at a paradigm shift, in which the *attributes* get a central role. There is an increasing demand to identify and authenticate citizens online and digitally exchange attributes related to their identities, such as their professional qualifications, driving licenses, age, and other permits or roles. At the same time, privacy concerns must be taken into account as well.

We have explored technical, usability, and privacy issues that come into play when extending the eIDAS network with more personal and domain-specific attributes. We have discussed the strength and limitations of the eIDAS MDS attributes. To retrieve and transfer additional attributes over the eIDAS network, we have provided technical solutions that have been implemented and assessed by involving students at our university. In general, students' perception was positive in the implemented academic services. They appreciated the

possibility to transfer personal identification data along with ''certified'' academic information over the trusted eIDAS network. Some of them provided helpful feedback about service usability, suggesting, for example, to further simplify the user consent pages.

We foresee future works in several parts. For example, self-sovereign identity models that give individuals control of their digital identities are emerging. The European Digital Identity framework [13] proposes to create digital wallets for this scope, where the user controls different types of attributes in different contexts based on her eID. We investigate how the proposed AP Connector can be exploited in the digital wallets implementation. We analyse how the eIDAS MDS can be further extended to support identity matching. Furthermore, we study mechanisms to derive privacy-preserving identifiers from the spidCode. Finally, concerning the domain-specific attributes, we believe further work is needed to define the semantics, the trust levels, and standardized ways to exchange them over the eIDAS nodes in production services.

ACKNOWLEDGMENT

The authors would like to thank our partners from Graz University of Technology, Austria, Universidad Politécnica de Madrid, Spain, Jozef Stefan Institute, Slovenia, and Universidade de Lisboa, Portugal, for their collaboration in the past eID4U project, where the roots of this work can be found. They would also like to thank Giorgio Santiano and Andrea Garzena from the Student Service Office of Politecnico di Torino, and Paolo Smiraglia from Agenzia per l'Italia Digitale (AgID), for their help to integrate the proposed solutions with the Politecnico di Torino backend and with the official Italian eIDAS node.

REFERENCES

[1] *Overview of the German Identity Card Project and Lessons Learned (2020 Update)*. Accessed: May 18, 2021. [Online]. Available: https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/inspired/eid-in-germany

[2] *Carta d'Identità Elettronica*. Accessed: May 18, 2021. [Online]. Available: https://www.cartaidentita.interno.gov.it/

[3] *11 Years of eID: Portugal's Citizen Card*. Accessed: May 18, 2021. [Online]. Available: https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/customer-cases/portugal-id

[4] *Portugal Lets Citizens Sign Documents With a Smartphone*. Accessed: May 10, 2020. [Online]. Available: https://joinup.ec.europa.eu/collection/joinup/news/digital-mobile-key

[5] *Infocert*. Accessed: May 18, 2021. [Online]. Available: https://www.infocert.it/

[6] *Poste Italiane*. Accessed: May 10, 2021. [Online]. Available: https://www.poste.it/

[7] *Sistema Pubblico di Identità Digitale*. Accessed: May 10, 2021. [Online]. Available: https://www.spid.gov.it/?lang=en-001

[8] ITU Report. *Digital Identity in the ICT Ecosystem: An Overview*. Accessed: May 18, 2021. [Online]. Available: https://www.itu.int/dms_pub/itu-d/opb/pref/D-PREF-BB.ID01-2018-PDF-E.pdf

[9] *Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market and Repealing Directive 1999/93/ec*, European Union, Brussels, Belgium, 2014.

[10] *eIDAS SAML Attribute Profile Version 1.2*. Accessed: May 10, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eIDAS%20SAML%20Attribute%20Profile%20v1.2%20Final.pd

[11] *eIDAS SAML Message Format Version 1.2*. Accessed: May 10, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eIDAS%20SAML%20Message%20Format%20v.1.2%20Final.pdf

[12] *How Can Identity Matching improve the Experience of Citizens on Online Public Services*. Accessed: May 18, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/2019/06/27/How+can+Identity+Matching+improve+the+experience+of+citizens+on+online+public+services

[13] *Commission Staff Working Document Impact Assessment Report Accompanying the Document Proposal for a Regulation of the European Parliament and of the Council amending Regulation (EU) No 910/2014 as Regards Establishing a Framework for a European Digital Identity*. Accessed: Jul. 13, 2021. [Online]. Available: https://op.europa.eu/en/publication-detail/-/publication/6f30628d-c458-11eb-a925-01aa75ed71a1/language-en

[14] A. Cavoukian. *Privacy by Design—The 7 Foundational Principles*. Accessed: May 18, 2021. [Online]. Available: https://iapp.org/resources/article/privacy-by-design-the-7-foundational-principles/

[15] S. Stalla-Bourdillon, H. Pearce, and N. Tsakalakis, ''The GDPR: A game changer for electronic identification schemes? The case study of Gov.UK verify,'' *Comput. Law Secur. Rev.*, vol. 34, no. 4, pp. 784–805, Aug. 2018, doi: 10.1016/j.clsr.2018.05.012.

[16] D. Hardt, *The OAuth 2.0 Authorization Framework*, document IETF RFC 6749, Oct. 2012.

[17] *JavaScript Object Notation*. Accessed: May 10, 2021. [Online]. Available: http://json.org/

[18] D. Berbecaru, A. Lioy, and C. Cameroni, ''Electronic identification for universities: Building cross-border services based on the eIDAS infrastructure,'' *Information*, vol. 10, no. 6, p. 210, Jun. 2019, doi: 10.3390/info10060210.

[19] D. G. Berbecaru, A. Lioy, and C. Cameroni, ''Providing login and Wi-Fi access services with the eIDAS network: A practical approach,'' *IEEE Access*, vol. 8, pp. 126186–126200, 2020, doi: 10.1109/ACCESS.2020.3007998.

[20] D. Berbecaru and C. Cameroni, ''ATEMA: An attribute enablement module for attribute retrieval and transfer through the eIDAS network,'' in *Proc. 24th Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Oct. 2020, pp. 532–539, doi: 10.1109/ICSTCC50638.2020.9259642.

[21] *Commission Implementing Regulation (EU) 2015/1501 of 8 September 2015 on th Interoperability Framework: Commission Implementing Regulation (EU) 2015/1502 of 8 September 2015 on Setting Out Minimu Technical Specifications and Procedures for Assurance Levels for Electronic Identification; Commission Implementing Decisio (EU) 2015/1984 of 3 November 2015 Defining the Circumstances, Formats and Procedures of Notification*, European Union, Brussels, Belgium, Nov. 2015.

[22] S. Lips, N. Bharosa and D. Draheim, ''eIDAS implementation challenges: The case of Estonia and The Netherlands,'' in *Electronic Governance and Open Society: Challenges in Eurasia, EGOSE 2020* (Communications in Computer and Information Science), vol. 1349, A. Chugunov, I. Khodachek, Y. Misnikov, and D. Trutnev, Eds, Cham, Switzerland: Springer, Nov. 2020, doi: 10.1007/978-3-030-67238-6_6.

[23] C. Satchell, G. Shanks, S. Howard, and J. Murphy, ''Beyond security: Implications for the future of federated digital identity management systems,'' in *Proc. 20th Conf. Comput.-Hum. Interact. Special Interest Group (CHISIG) Aust. Comput.-Hum. Interact., Design: Activities, Artefacts Environ. (OZCHI)*, 2006, pp. 313–316, doi: 10.1145/1228175.1228231.

[24] L. Bauer, C. Bravo-Lillo, E. Fragkaki, and W. Melicher, ''A comparison of users' perceptions of and willingness to use Google, Facebook, and Google+ single-sign-on functionality,'' in *Proc. ACM Workshop Digit. Identity Manage.*, Nov. 2013, pp. 25–36, doi: 10.1145/2517881.2517886.

[25] H. Gomi, ''User-centric identity governance across domain boundaries,'' in *Proc. 5th ACM Workshop Digit. Identity Manage. (DIM)*, 2009, pp. 35–44, doi: 10.1145/1655028.1655038.

[26] *MyData Global*. Accessed: Jul. 13, 2021. [Online]. Available: https://mydata.org/

[27] *Commission Staff Working Document Accompanying the Document Report From the Commission to the European Parliament and the Council on the Evaluation of Regulation (EU) No 910/2014 on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market (eIDAS)*, Accessed: Jul. 13, 2021. [Online]. Available: https://op.europa.eu/en/publication-detail/-/publication/00dfddeb-c449-11eb-a925-01aa75ed71a1/language-en

[28] N. Taniguchi, K. Chida, O. Shionoiri, and A. Kanai, "DECIDE: A scheme for decentralized identity escrow," in *Proc. Workshop Digit. Identity Manage. (DIM)*, 2005, pp. 37–45, doi: 10.1145/1102486.1102493.
[29] A. Bhargav-Spantzel, A. C. Squicciarini, and E. Bertino, "Establishing and protecting digital identity in federation systems," *J. Comput. Secur.*, vol. 14, no. 3, pp. 269–300, Jun. 2006, doi: 10.3233/JCS-2006-14303.
[30] M. S. Ferdous and R. Poet, "Analysing attribute aggregation models in federated identity management," in *Proc. 6th Int. Conf. Secur. Inf. Netw. (SIN)*, 2013, pp. 181–188, doi: 10.1145/2523514.2526998.
[31] A. Jøsang, J. Fabre, B. Hay, J. Dalziel, and S. Pope, "Trust requirements in identity management," in *Proc. Australas. Inf. Secur. Workshop (AISW), Conf. Res. Pract. Inf. Technol.*, vol. 44, P. Montague and R. Safavi-Naini, Eds. Newcastle, NSW, Australia, Feb. 2005, pp. 99–108. Accessed: Sep. 28, 2021. [Online]. Available: https://crpit.scem.westernsydney.edu.au/confpapers/CRPITV44Josang.pdf
[32] *eID4U Project*. Accessed: May 10, 2021. [Online]. Available: https://ec.europa.eu/inea/en/connecting-europe-facility/cef-telecom/2017-eu-ia-0051
[33] DE4A (Digital Europe for all) Project. *Studying Abroad Use cases Definition and Requirement*. Accessed: May 10, 2021. [Online]. Available: https://www.de4a.eu/project-deliverables
[34] D. Berbecaru and A. Lioy, "On integration of academic attributes in the eIDAS infrastructure to support cross-border services," in *Proc. 22nd Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Oct. 2018, pp. 691–696, doi: 10.1109/ICSTCC.2018.8540674.
[35] HEALTHeID eIDAS OpenNCP Connector for eHealth. *D2.1. HEALTHeID Functional Specification*. Accessed: May 10, 2021. [Online]. Available: https://spms.min-saude.pt/wp-content/uploads/2019/08/D2.1_HealtheID_functionalSpecs_20190523_rev.pdf
[36] *MyAcademicID Project*. Accessed: May 10, 2021. [Online]. Available: https://myacademic-id.eu/
[37] (Sep. 23, 2020). *MyAcademicID Blueprint Architecture, Version 1.0*. Accessed: May 10, 2021. [Online], Available: https://myacademic-id.eu/images/MyAID_Blueprint_Architecture_Final.pdf
[38] *Mobile Cross-Border Government Services for Europe (mGov4EU)*. Accessed: May 10, 2021. [Online]. Available: https://www.mgov4.eu/index.html
[39] D. Berbecaru, A. Lioy, and C. Cameroni, "Providing digital identity and academic attributes through European eID infrastructures: Results achieved, limitations, and future steps," *Softw., Pract. Exper.*, vol. 49, no. 11, pp. 1643–1662, Nov. 2019, doi: 10.1002/spe.2738.
[40] D. Berbecaru and A. Lioy, "On the design, implementation and integration of an attribute provider in the pan-European eID infrastructure," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2016, pp. 1263–1269, doi: 10.1109/ISCC.2016.7543910.
[41] *What can eID do for You*. Accessed: May 10, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eID+for+You
[42] *eIDAS Cryptographic Requirements for the Interoperability Framework, TLS and SAML, Version 1.0*. Accessed: May 10, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eidas_-_crypto_requirements_for_the_eidas_interoperability_framework_v1.0.pdf?version=1&modificationDate=1497252920224&api=v2
[43] E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.3*, document RFC 8446, Aug. 2018, doi: 10.17487/RFC8446.
[44] D. Berbecaru, A. Atzeni, M. De 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)*, 2017, pp. 391–395, doi: 10.1109/PDP.2017.90.
[45] *EU: Commission Implementing Regulation (EU) 2015/1501 of 8 September 2015 on the Interoperability Framework Pursuant to Article 12(8) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market*. Accessed: May 10, 2020. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A02015R1501-20150909
[46] N. Tsakalakisz, S. Stalla-Bourdillon, and K. O'Hara, "Identity assurance in the UK: Technical implementation and legal implications under eIDAS," *J. Web Sci.*, vol. 3, no. 1, pp. 32–46, Dec. 2017, doi: 10.1561/106.00000010.
[47] *eIDAS Interoperability Architecture Version 1.2*. Accessed: May 10, 2020. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eIDAS%20Interoperability%20Architecture%20v.1.2%20Final.pdf
[48] *New Notified eID Schemes in 2020*. Accessed: May 10, 2020. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/2020/12/17/ New+notified+eID+schemes+in+2020
[49] *Overview of Pre-Notified and Notified eID Schemes Under eIDAS*. Accessed: May 10, 2020. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/display/EIDCOMMUNITY/Overview+of+pre-notified+and+notified+eID+schemes+under+eIDAS
[50] *Electronic ID cards in Belgium: The keystone of eGovernment*. Accessed: May 10, 2020. [Online]. Available: https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/customer-cases/belgium
[51] *Share of Households With an eID Card Reader in Belgium 2008–2018*. Accessed: May 10, 2020. [Online]. Available: https://www.statista.com/statistics/558946/share-of-households-with-an-eid-card-reader-in-belgium/
[52] *Itsme.Be English Blog*. Accessed: May 10, 2020. [Online]. Available: https://www.itsme.be/en/blog
[53] *Next Generation NemID*. Accessed: May 10, 2020. [Online]. Available: https://en.digst.dk/digitisation/eid/next-generation-nemid/
[54] *e-Estonia.Com E-Identity*. Accessed: May 10, 2020. [Online]. Available: https://e-estonia.com/solutions/e-identity/id-card/
[55] *Estonian Information System Authority Means of eID*. Accessed: May 10, 2020. [Online]. Available: https://e-estonia.com/solutions/e-identity/id-card/
[56] *Avanzamento Trasformazione Digitale, SPID*. Accessed: May 10, 2020. [Online]. Available: https://avanzamentodigitale.italia.it/it/progetto/spid
[57] *Public Broadcasting of Latvia, eID Cards to Become Mandatory Identification Documents in 2023*, Accessed: May 10, 2020. [Online]. Available: https://eng.lsm.lv/article/society/society/eid-cards-to-become-mandatory-identification-documents-in-2023.a290382/
[58] *LuxTrust, What Exactly is an Electronic Identity (eID)*. Accessed: May 10, 2020. [Online]. Available: https://www.luxtrust.com/what-exactly-is-an-electronic-identity-eid/
[59] (2014). *The Year of the Slovakian eID*. [Online]. Available: https://silicontrust.org/2014/03/31/2014-the-year-of-the-slovakian-eid/
[60] *Slovakia Uses eID Card for Safe Digital Public Services*. Accessed: May 10, 2020. [Online]. Available: https://thrive.dxc.technology/eur/2019/05/16/slovakia-uses-eid-card-for-safe-digital-public-services/
[61] *Spanish ID Cards, Evolution and Meaning of DNI 3.0 Fields*. Accessed: May 10, 2020. [Online]. Available: https://www.mobbeel.com/en/blog/spanish-id-cards-evolution-and-meaning-of-dni-3-0-fields/
[62] *Digital Government Factsheet 2019—The Netherlands*. Accessed: May 10, 2020. [Online]. Available: https://joinup.ec.europa.eu/sites/default/files/inline-files/Digital_Government_Factsheets_Netherlands_2019_0.pdf
[63] U. B. Mir, A. K. Kar, Y. K. Dwivedi, M. P. Gupta, and R. S. Sharma, "Realizing digital identity in government: Prioritizing design and implementation objectives for Aadhaar in India," *Government Inf. Quart.*, vol. 37, no. 2, Apr. 2020, Art. no. 101442, doi: 10.1016/j.giq.2019.101442.
[64] Á. Alonso, A. Pozo, A. Gordillo, S. López-Pernas, A. Munoz-Arcentales, L. Marco, and E. Barra, "Enhancing university services by extending the eIDAS European specification with Academic attributes," *Sustainability*, vol. 12, no. 3, p. 770, Jan. 2020, doi: 10.3390/su12030770.
[65] T. Klobučar, "Facilitating access to cross-border learning services and environments with eIDAS" in *Learning and Collaboration Technologies. Ubiquitous and Virtual Environments for Learning and Collaboration* (Lecture Notes in Computer Science), vol 11591, P. Zaphiris and A. Ioannou, Eds. New Delhi, India: HCII, 2019, doi: 10.1007/978-3-030-21817-1_25.
[66] *FICEP First Italian Crossborder eIDAS Proxy Server*. Accessed: Sep. 28, 2021. [Online]. Available: https://www.eid.gov.it/presentazione-progetto?lang=en-001
[67] *Agenzia Per l'Italia Digitale*. Accessed: May 10, 2021. [Online]. Available: https://www.agid.gov.it/
[68] (in Italian). *SPID Registry*. Accessed: May 10, 2021. [Online]. Available: https://registry.spid.gov.it/
[69] *SPID Regole Tecniche*. Accessed: May 10, 2021. [Online]. Available: https://docs.italia.it/italia/spid/spid-regole-tecniche/it/stabile/attributi.html
[70] *eIDAS Node Version 1.4.4 Source Code*. Accessed: May 10, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eIDAS-Node+version+1.4.4
[71] *eIDAS-Node Installation Manual v1.4.4*. Accessed: Sep. 28, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/download/attachments/84421967/eIDAS-Node%20Installation%20Manual%20v1.4.4.pdf
[72] *Apache Struts 2*. Accessed: May 10, 2021. [Online]. Available: https://struts.apache.org/

[73] A. Rodriguez. (2008). *Restful Web Services: The Basics*. IBM developerWorks. Accessed: May 10, 2021. [Online]. Available: https://cs.calvin.edu/courses/cs/262/kvlinden/references/rodriguez-restfulWS.pdf
[74] *NGINX Reverse Proxy*. Accessed: Sep. 28, 2021. [Online]. Available: https://docs.nginx.com/nginx/admin-guide/web-server/reverse-proxy/
[75] *NGINX SSL Termination*. Accessed: May 10, 2021. [Online]. Available: https://docs.nginx.com/nginx/admin-guide/security-controls/terminating-ssl-http/
[76] NGINX SSL Termination, Accessed: May 10, 2021. [Online]. Available: https://docs.nginx.com/nginx/admin-guide/security-controls/terminating-ssl-http/
[77] *Jakarta EE*. Accessed: Sep. 28, 2021. [Online]. Available: https://jakarta.ee/
[78] *Eclipse MicroProfile Project*. Accessed: May 10, 2021. [Online]. Available: https://projects.eclipse.org/projects/technology.microprofile
[79] Eclipse MicroProfile Project, Accessed: May 10, 2021. [Online]. Available: https://projects.eclipse.org/projects/technology.microprofile
[80] *Open Liberty Project*. Accessed: May 10, 2021. [Online]. Available: https://openliberty.io/
[81] *Nimbus JOSE + JWT Library*. Accessed: May 10, 2021. [Online]. Available: https://connect2id.com/products/nimbus-jose-jwt
[82] *OpenID Connect*. Accessed: May 10, 2021. [Online]. Available: https://openid.net/connect/
[83] *OpenID Connect Explained*. Accessed: May 10, 2021. [Online]. Available: https://connect2id.com/learn/openid-connect
[84] M. Jones, J. Bradley, and N. Sakimura, *JSON Web Token (JWT)*, document RFC 7519, IETF, May 2015, doi: 10.17487/RFC7519.
[85] *Jakarta EE 8 Specification APIs—The Client API*. Accessed: May 10, 2021. [Online]. Available: https://jakarta.ee/specifications/platform/8/apidocs/javax/ws/rs/client/package-summary.html
[86] *Eclipse MicroProfile JWT RBAC Security (MP-JWT)*. Accessed: May 10, 2021. [Online]. Available: https://www.eclipse.org/community/eclipse_newsletter/2017/september/article2.php
[87] *CEF Digital, eIDAS-Node Integration Package*. Accessed: Sep. 28, 2021. [Online]. Available: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eIDAS-Node+Integration+Package

DIANA GRATIELA BERBECARU received the Ph.D. degree in computer engineering from the Politecnico di Torino, Italy, and the M.Sc. degree in computer science from the University of Craiova, Romania. She is currently a Senior Research Assistant with the Department of Control and Computer Engineering, Politecnico di Torino. She is also a member of the TORSEC Cybersecurity Research Group. Her research interests include digital certificates, network forensics, multi-cast authentication, and electronic identification.

ANTONIO LIOY received the M.Sc. degree *(summa cum laude)* in electronic engineering and the Ph.D. degree in computer engineering from the Politecnico di Torino. He is currently a Full Professor with the Politecnico di Torino, where he leads the TORSEC Cybersecurity Research Group. His research interests include network security, policy-based system protection, trusted computing, and electronic identity.

CESARE CAMERONI received the M.Sc. degree in computer engineering from the Politecnico di Torino, in 2010. From 2011 to 2014 and since 2018, he has been a Research Assistant with the Department of Control and Computer Engineering, Politecnico di Torino. He is currently a member of the TORSEC Cybersecurity Research Group. His research interests include security and dependability of ICT systems, electronic identities, and federated identity infrastructures.