European
Commission

J R C T E C H N I C A L R E P O R T S

Electronic Soft-Identities (E-Ids)

State-of-the-art and Multi-Morphed E-Ids, an Explorative Study

Igor NAI FOVINO

Ricardo NEISSE

Dimitris GENEIATAKIS

Ioannis KOUNELIS

2013

Report EUR 26523 EN

Joint
Research
Centre

European Commission
Joint Research Centre
Institute for the Protection and Security of the Citizen

Contact information
Igor Nai Fovino
Address: Joint Research Centre, Via Enrico Fermi 2749, TP 361, 21027 Ispra (VA), Italy
E-mail: igor.nai-fovino@jrc.ec.europa.eu
Tel.: +39 0332 785809
Fax: +39 0332 785145

http://www.jrc.ec.europa.eu/
http://www.ipsc.jrc.ec.europa.eu/

This publication is a Reference Report by the Joint Research Centre of the European Commission.

JRC87768

EUR 26523EN

ISBN 978-92-79-35529-5

ISSN 1831-9424

doi 10.2788/78078

Luxembourg: Publications Office of the European Union, 2014

Printed in Ispra

Table of Contents

1 Executive Summary

The online identity of the end-user is one of the key enabler of every modern digital service. To get the access to the cloud, to perform e-commerce transactions, to chat online and to perform many other operations, the user needs to provide proof of his digital identity (e-id). In doing that he expose himself to the risk of disclosing sensitive information on his life and his online activities.

The scientific contribution of this report is twofold: on a side it provides a first, explorative overview of the state of the art in the world of soft-identity management systems, and on the other he presents the first outcomes of a research activity aiming at proposing a solution to address trust and privacy protection issues related to identity and personal data provided by citizens in a smart environment. The proposed solution combines identity management, trust negotiation, and usage control. The concept of identity management allows creation of less privacy sensitive soft identities derived from hard identities with high assurance. Trust negotiation techniques are used during the authentication phase to support the identity establishment process between the entities in the smart city. After the identity is established we use usage control policies to govern the exchange of identity and personal data in a privacy friendly manner.

This report constitutes the first of a series of reports which aim at defining methods and techniques to empower the citizen in the protection of his online privacy.

2 Introduction

The role of identity is extremely important in our society. On the basis of our identities we are allowed or denied to perform every day vital operations.

From a philosophical point of view, the identity is the key of every human interaction. We adapt the interaction with a person on the basis of an evaluation of his identity and the surrounding context.

We accept to execute tasks on the basis of the identity of the person requiring that task; we trust on information obtained on the basis of the identity of the information source.

Traditionally the evaluation of the identity of a person involves information related to:

1. What we know about this person
2. What we see and feel about this person
3. What others say about this person (being the “others” provided with some level of trust)

Within the whole “game” of evaluating an identity, establishing a level of trust and acting in consequence, a strong role is played by the possibility of physically verifying who the counterpart with whom we are interacting is.

In the digital world, on the other hand, human interactions are indeed extremely limited and the identity evaluation relies obviously less on point (2) and a more on point (1) and (3).

According to the standard ISO/IEC 24760 (part 1), a digital identity is defined as “a set of attributes related to an entity”, where entity refers to an individual, an organization, or a device. Attributes are properties of the entity (e.g. address, phone number etc.).

Digital identities can be categorized according to the security level adopted in the registration and authentication phases, i.e. when a digital identity is associated to a target entity. So we can have Hard and Soft electronic identity (e-id).

In our digital society, the concept of digital identity is becoming more and more relevant and in fact, the section 2.1.2 of the “Digital Agenda for Europe” makes an explicit reference to digital identities:

“Electronic identity (eID) technologies and authentication services are essential for transactions on the Internet both in the private and public sectors. […] As there will be many solutions, industry, supported by policy actions – in particular eGovernment services - should ensure interoperability based on standards and open development platforms”.

The problem is that, outside the realm of the so called Strong-eID (e.g. electronic ID cards), the average citizen does not pay enough attention to its his digital identity, and, in several cases, he is not even aware of possessing one, or, more commonly, multiple identities.

An e-mail account is a digital identity, the account I use to write on a forum is a digital identity as well as Facebook, Dropbox, Twitter, and PayPal accounts are.

The fact is that a single format for our online identities does not exist, as a set of unified procedures regulating their protection and management is not defined. As specified in ISO/IEC 24760, everything which can be used to identify myself online in a unique manner is, per se, a digital identity.

In this report we are focusing on soft identities security issues as this kind of identities are the ones mostly used by the citizens in their daily activities. Accessing social networks, email accounts and even watching programs from a smart television requires a soft identity. Moreover, due to their frequent use the user may not realize the importance or the implications of misusing or losing one. In most cases it happens that such identities are interlinked either in an immediate or intermediate way. For example, a Facebook identity is linked with one user email identity which is linked with a second email identity as a backup solution and so on. In the end it turns out that a chain of identities from the same physical person exists, and if one of the bonds is vulnerable, all the other bonds may become vulnerable as well. This is in fact not rare as one can see from incidents like this: [http://www.wired.com/gadgetlab/2012/08/apple-amazon-mat-honan-hacking/all/]

2.1 Digital Identity in the IoT and Smart World

The digital identity definition has been extended recently with a sort of “inheritance principle”.

Citizens are starting to make massively use of smart-devices and smart-sensors which are connected to the Internet.

To get access to online services they need to configure their devices using their own credentials, giving to these devices rights to operate in their name.

Let take as example a smart-TV: the citizen, to download and see content should provide to the smart-TV a mean to authenticate itself to the online services. Typically, the authentication will imply the use of some sort of digital-identity linked to the owner of the TV-subscription; in other words, the smart-TV inherits a “portion” of the identity of its owner. The same situation happens when for example the citizen configures his mobile-phone to get synchronized with the company’s calendar. To get direct access to this commodity, the smart-phone will need to authenticate itself to the calendar service using some personal credentials; again, the smart-device inherits part of the identity of its owner.

The same principle can be applied considering the more extended scenario of a Smart City, where digital identities or aggregates of digital identities are associated to complex systems used to deliver secure and trusted physical services to the citizen, e.g. public transportation, car to car communication, remotely monitored Health care devices etc.

However, digital identities do not impact only on the daily life of the citizen, as their role is becoming more and more important also in the industrial sector.

Lets consider the world of Industrial Control Systems (ICS); the increasing use of general purpose telecommunication networks (i.e. Internet) in these infrastructures, acted as a sort of glue, so that, today, we can say that ICS (and SCADA systems) are remotely controlled and accessed. Also in this case digital identities have a relevant role. To access to certain remote components or control servers, identities with associated roles and rights need to be used. Their management, the way in which they are protected and revoked – if needed, should and must be one of the top priorities for the security of a critical infra-structure.

The same consideration can be done also when thinking about the communication of low level control devices (e.g. PLCs). In this case, especially for those installations spread in geographically remote locations, with scarce or in-existing surveillance (let consider for example a gas or oil pipeline passing through remote regions of the world), the problem of securely manage their digital identities (in this case crypto-material allowing to sign and authenticate their readings and control messages) should be of high relevance.

An interesting playground where citizen identities and industrial infrastructures are quickly converging is that of smart-metering. Smart-meters can be considered the ultimate leafs of the smart-grids. These objects are at the moment those in charge for measuring the energy consumptions of the citizen, and, in some countries, for measuring also the energy production of the citizen.

However, to really benefit from the establishment of a smart-energy grid, soon these meters will need to get more and more integrated, on a side, with the energy-distribution infrastructure, and on the other, with the citizen's home digital infrastructure. Here again the digital identity inheritance principle described before will play a relevant role in the protection of the privacy of the citizen while guaranteeing the provisioning, in a secure way, of services allowing to improve the optimization of the energy consumption and production.

2.2 Soft Digital Identity Challenges

The concept of digital identity acted, as stated before, as enabler to get the access to a huge amount of different online services. However, a digital identity is also a possible key to get access to a huge amount of citizen's personal information and might be subject to profiling analysis from which additional information on the e-ID owner can be derived. This is especially true for the so called soft-identities, which are, by definition and nature, not standardized and to which, normally, the citizen pay poor attention in term of security despite the fact that they are commonly used indeed to access an incredible amount of personal information (think about the account of a social network).

From what briefly presented before, we can say that the infrastructures managing the digital identities will become more and more critical for the security and privacy of the citizen.

Under this light, generally speaking, three are the real challenges and needs:

1. Provide the citizen with means to control and regulate the use of the sensitive information made accessible through a certain soft-digital identity
2. Identify the right trade-off between level of disclosure (i.e. the amount of information associated to a certain digital identity when used) and the citizen's privacy level. This point assumes a high relevance especially in the context of digital identity inheritance, where smart-devices uses some piece of their owner's identity to autonomously interact with the external digital world
3. Educate the digital citizen to a better use of their digital identities

Only in this way it would be possible to establish a correct level of trust in the digital world.

This report is organized as follows: in Section 3 is provided a first overview of digital identity related concepts, while in Section 4 are briefly described the standardization effors. Section 5 provides a description of the most used protocols and architectures dealing with identity management while Section 6 presents a classification of existing identity management research solutions. After analyzing in Section 7 the issues related to digital identity management, in Section 8 a framework allowing to enforce the control of the citizen on his soft-identities and personal related information is described. The prototype implementing the framework is showed in Section 9, while Section 10 provide a brief overview of the test-be deployed to analyze the impact of the cloud on the security and privacy of the citizen. In Section 11 conclusions are presented.

3 Identity Models and Architectures

As claimed in the introduction, the definition of the e-ID concept is not trivial, as, being linked to personal information, involve subjective elements.

3.1 Electronic Identity (e-Id) Definition

A real-world entity is represented in the digital world by an Electronic Identity (e-Id), which consists of a set of attributes associated to this entity. Identity attributes are a collection of name/value pairs as Figure 1 illustrates.

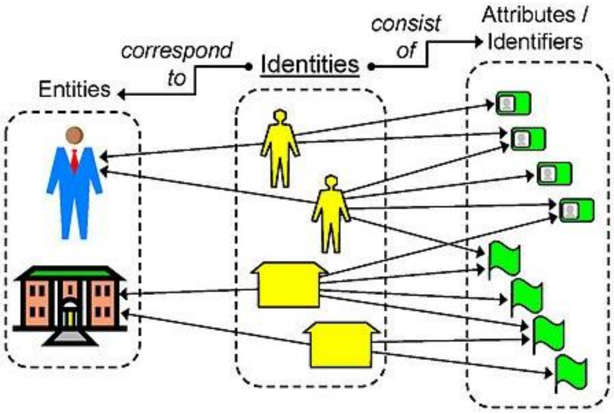


Figure 1 – Digital Identity Model (source [15])

According to the standard ISO/IEC 24760 (part 1) [43] a digital identity is defined as *"a set of attributes related to an entity"*, where *entity* refers to an individual, an organization, or a device.

Attributes are properties of the entity (e.g. address, phone number etc.), and they can be unique, i.e. able to identify completely their owner, or can be more generic, i.e. referred to the owner, but not able to provide a unique characterization of him/her.

As in the case of physical identities (e.g. ID cards, passports, driving licenses etc.), also in the realm of digital identities an important role is occupied by the process of ID issuing.

Under this light, digital identities can be categorized according to the security level adopted in the registration and authentication phases, i.e. when a digital identity is associated to a target entity. So we can have **Hard** and **Soft** electronic identities (e-ids). In the following Figure an example of identity classification is provided.

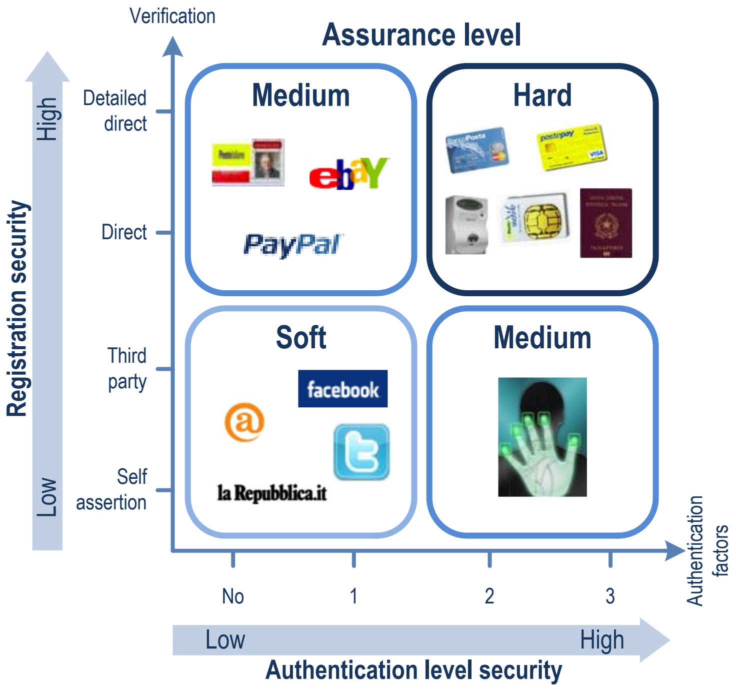


Figure 2 – Example Identity Classification

In addition, not only the registration phase is relevant when evaluating the security, robustness and trust of a digital identity, but also the whole process of identity management.

In this context, ISO/IEC 24760-2 clearly identifies a set of relevant roles:

- **Principal**: Entity to which identity information pertains
- **Identity management authority:** Entity associated with a domain with the capabilities to set and enforce operational policies
- **Relying party:** Entity that relies on the verification of identity information for a particular entity
- **Identity Information Authority:** Entity related to a particular domain that can make provable statements on the validity and/or correctness of one or more attribute values in an identity
- **Identity Information Provider:** Entity related to a particular domain that can make provable statements on the validity and/or correctness of one or more attribute values in an identity
- **Verifier:** Entity that performs verification
- **Auditor**: Entity with capabilities to inspect operations

Despite the fact that ISO standards clearly define all the relevant elements of the digital identity realm, their presence into real frameworks is not always implemented, especially when considering soft identities. Unfortunately this fact constitutes, per se, a problem: since by definition in soft identities the registration phase is weaker, less regulated, the full implementation of all the elements defined by the ISO standard would greatly enhance the level of security of soft-IDs.

3.1.1 Areas of Application

Digital identity solutions are currently applied in the following areas:

- **Government level:** Passports, Identity cards, Driving Licenses, public health insurance cards.
- **Corporate level:** customer access, bank accounts, credit cards.
- **Personal:** personal/professional (Business services) sign-on, financial services, social networks (Facebook), business networks (LinkedIn), and many cloud services.

3.1.2 Identity Management Architectures

Identity management architectures can be classified considering the responsible entity for managing the users' credentials and identifiers into: isolated, federated, centralized, and personal [15]. In the isolated architecture the user identities are managed independently by the service providers. In the federated architecture identifiers are shared between a set of service providers after an agreement is established between them, which is equivalent to a Single-Sign-On (SSO) approach. In a centralized approach, credentials and identifiers are managed by a third party (identity provider), and users may use the same identifier for all service providers (common identifier domain) or different identifiers (meta-identifier domain) to prevent tracking of their activities. Finally, users may decide to manage their identities for each service provider themselves, and this type of architecture is called a personal identity provider.

Different identity providers may collaborate in an identity federation to enable identities provided by one identity provider to be automatically recognized and trusted by other identity providers. Figure 3 shows an example of an identity federation with users represented in blue, identity providers in green, and service providers (a.k.a. relying parties) in gray. The blue ellipse associates the users with an identity provider and the grey ellipse the identity providers with a set of service providers. The green ellipse represents a federation between identity providers.

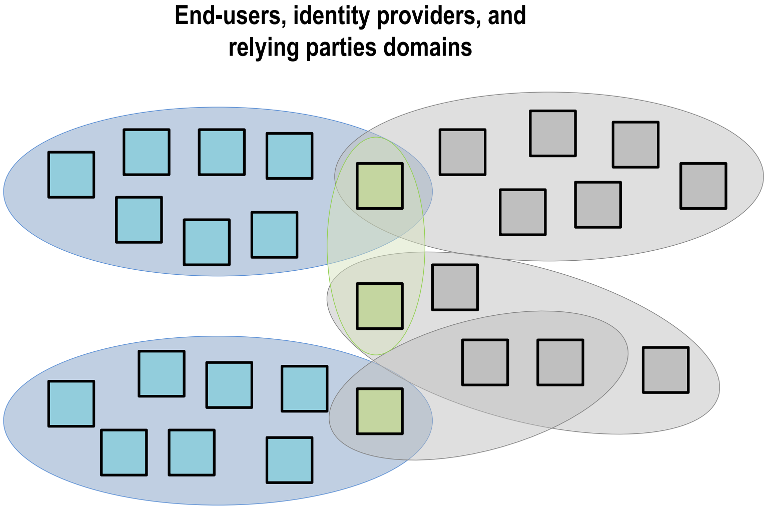


Figure 3 Example of Identity Federation and SSO

4 Standardization Efforts

The **ISO/IEC 24760** suite of standards addresses the matter of identity principally from a management perspective. In particular ISO 24760-1 presents a reference for identity management, and specifies relevant concepts of identity and identity management and their relationships. ISO 24760 part 2 and 3 are still in their drafting phase (final version expected by 2015), but they are expected to propose a reference architecture and a set of requirements related to the identity management (part 2), and a set of practices (part 3).

ISO/IEC 29100:2011 *"Information technology -- Security techniques -- Privacy framework"* defines the actors and their roles in processing personally identifiable information (PII). Aspects specified in this standard, should be taken into consideration when defining the minimum privacy and security requirements of digital identity (both from a design and management point of view).

Other standards-practices-recommendations of interest:

- **NIST SP 800-63:** it defines (a) the guidelines for user's remote authentication and (b) the technical requirements for the assurance levels of proofing, registration, tokens and authentication protocols.
- **ISO/IEC 29115:** it defines the concept of entity authentication assurance framework
- **ISO/IEC 29146:** it defines a framework for access management
- **ISO/IEC 27001:** it provides the floor for the definition of an Information Security Management System (ISMS), which is relevant for the management of digital identity systems.
- **ISO/IEC 9798:** it provides, among other things, an authentication model
- **ISO/IEC 29100:** it defines a Privacy Framework and the key actors and principles for the management of personal data (included then the digital identities)
- **ISO/IEC 29100:** it defines a Privacy Framework and the key actors and principles for the management of personal data (included then the digital identities)
- **ISO/IEC 29101:** under development, it will provide a Privacy reference architecture framework
- **ITU-T X.1252:** it defines a baseline for identity management terms and definitions
- **ITU-T Y.2720:** NGN identity management framework
- **ITU-T X.1251:** A framework for user control of digital identity

5 Protocols and Technical Approaches

In the industry world a set of standards, protocols and technical approaches have been developed to support the different required functionalities in the identity management solutions. Though several approaches have been proposed in literature [37], we describe in the following only three of them, i.e. OpenID [30], OAuth [31], and Kerberos [32] since these solutions have been incorporated in products, such as Windows OS, Facebook, Google Wallet, etc., which are used by numerous users. Thanks to their large coverage, they can be intended as didactic means to understand the general security implications of every identity management solution. Furthermore, the understanding of those solutions should be considered of high importance as we are moving on cloud architectures that might introduce additional threats from the users' perspective.

5.1 OpenID

OpenID [30] is an open standard to support authentication delegation to third party identity providers. Using OpenID users authenticate through a trusted third party *identity service* before accessing the wished service. OpenID supports the implementation of Single-Sign-On (SSO) and identity federations as well. In this way, users do not need to register for every service that they would like to use, but they can use the same account/credential to access different web services. In this architecture, the service provider is not needed to manage users' identities and the corresponding credentials.

When first interacting with an OpenID-enabled cloud service, an user needs to select an identity provider to use from a list of supported OpenID providers and to submit the claimed identity to the service. This obviously implies that previously the user managed to obtain an account/credential/identity from the identity provider. After providing this information, the user is redirected to the identity provider to perform the authentication process, using his service provider's identity claim. The authentication process consists of two steps: identification and verification of the identity using the appropriate credentials (e.g. username/password). If the authentication is successful the identity provider redirects the user back to the service provider with an authentication token, which includes among others the user's verified identity and the identity provider signature. The service provider extracts the authentication token to securely verify it and retrieve if required additional identity information from the identity provider. The verification procedure relies on a shared key, if established, otherwise the service provider forwards the authentication token to the OpenID provider for the appropriate validation. In the case of establishing a shared key between the OpenID and the Service provider there are two options:

1. No encryption
2. Diffie Hellman (DH) [33]

If the "no encryption" mode is selected the shared key can be manipulated constituting the service vulnerable to impersonation attacks. In the case of DH the service chooses the DH parameters p, g. According to the OpenID specifications the default value for g is 2. The service provider chooses a random number R_s and sends the value of K_s =g^ R_s mod p to the OpenID provider. The OpenID provider chooses the shared key and computes the encryption key by choosing a random number R_p

by using the following formula $K_{enc} = K_s \wedge R_p \mod p$. The OpenID provider returns to the service provider the encrypted shared key, which its value is computed by the following formula $SHA(K_{enc})$ *XOR secret* and the $K_p = g \wedge R_p \mod p$, which is used by the latter to compute the encryption shared key by using the following formula $K_{enc} = K_p \wedge Rs \mod p$ for extracting the *shared key.* This shared key is used to validate the authentication token. It should be noted that in this setup users' identities are exchanged in clear text.

After the authentication token validation, the user can specify an authorization policy stating which identity profile information the service provider is allowed to access. This procedure is illustrated in the following Figure.

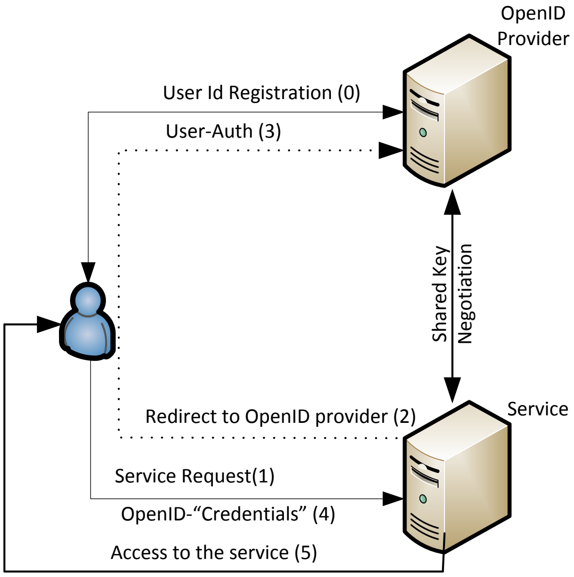


Figure 4. OpenID User's Identity Verification Procedure

5.2 OAuth

Authorization is the decision process after the authentication to allow or deny access to a resource. One prominent authorization standard currently supported by many service providers is OAuth [31]. Using OAuth users (resource holders) can temporarily allow access to their resources using an access token to a given third party service, without having to share their credentials with the latter. This is similar to the procedure followed in OpenID for validating users' identities. An example application of OAuth is a user that would like to allow access to his calendar managed by a cloud service (e.g. Google Calendar) to a friend.

In the OAuth framework, a user provides to a service an authorization grant in order to obtain access to the requested resources. To do this, the service requests to the end user to give an authorization access to the requested resources. The user is authenticated to the OAuth framework which issues the authorization token and sends it back to the service via the user. This procedure relies on HTTP Authentication [ref], while the authorization token is computed based on [HTTP Authentication: MAC Access Authentication]. The service in order to get access to the resource provides the authorization token and requests an access token from the resources server. The access token afterwards is used for accessing the requested resource. Note that in order to protect the exchanged information it is suggested to employ the OAuth over Transport Layer Security (TLS). The whole procedure of OAuth is illustrated in the following Figure.

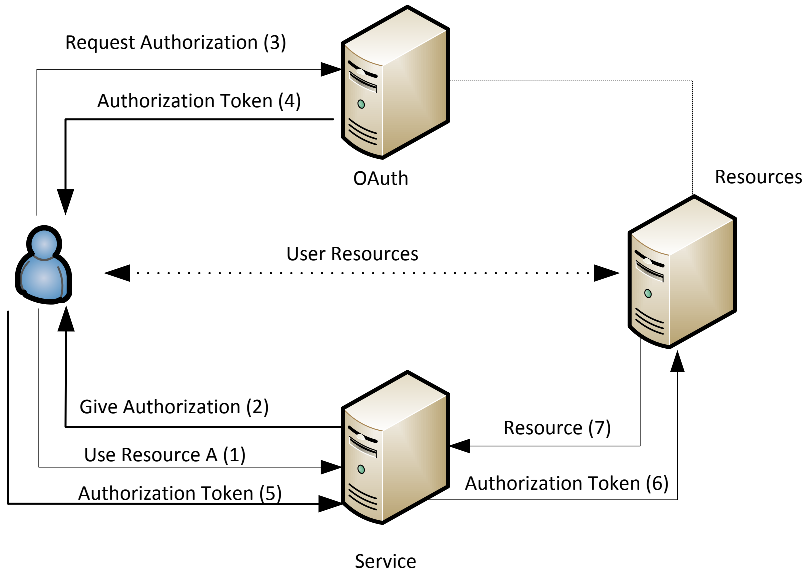


Figure 5. OAuth Authorization Procedure

5.3 Kerberos

Kerberos is a network device mutual authentication protocol developed originally by the Massachusetts Institute of Technology (MIT) [1]. Kerberos requires a trusted third party to manage user passwords, public keys, and issue tickets. This trusted third party is called the Key Distribution Center (KDC), and consists of the Authentication Service (AS) and Ticket Generation Service (TGS).

Using Kerberos, entities can authenticate securely on open unprotected network, assuming attackers have full control and can read, modify, and insert network packets. The client is authenticated with the KDC using a set of credentials, which might be a username/password combination. A result of the authentication process is also a session ticket that allows the verification of the client to other services that belong to the same domain. More specifically, whenever a client would like to establish trust with a specific service he generates an authentication request towards the KDC. The request contains the user's identity, service name, expiration time for the authentication token and a random number. The KDC authenticates the client and generates a Ticket Granting Ticket (TGT), which is used by the client to retrieve tickets for accessing other services, and a session key. The TGT is encrypted with the KDC secret key, while the session key is encrypted with the users' secret key. To access the service the client should first obtain a Ticket Grant Service from the KDC by sending a request that includes the TGT and an authenticator generated by the client and encrypted with the session key. The KDC responds to that request with a ticket, which is encrypted with the service secret key and a service session key generated by TGS and encrypted using the previous session key. As soon as the client receives the service ticket he can request to access the service. This is accomplished by generating another request towards the service in which is included the ticket and an authenticator again generated by the client, but this time it is encrypted using the service session key. The service checks the request and if required by the client sends a response to accomplish mutual authentication. The high level procedure is illustrated in Figure 6, while the details of the protocol messages are presented in Table 1.

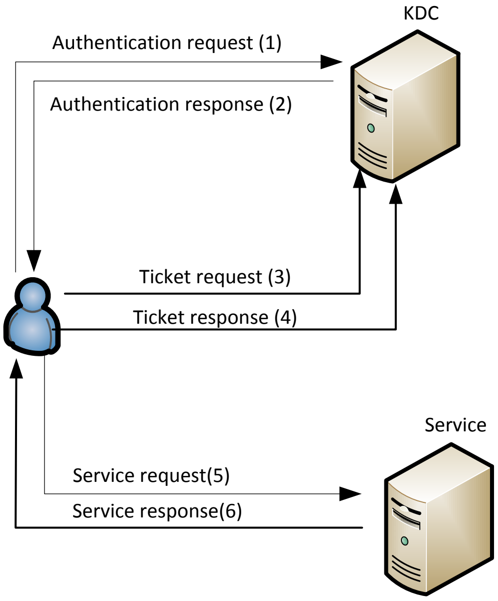


Figure 6. Kerberos authentication procedure.

Message	Descrition
`(1){UserId, ServiceId, Expr}`	The authentication request contains the user and the service id and a suggested lifetime for the ticket that will be generated.

`(2){UserId,Serviceid,TimeStamp,Expir,SKKdc}Kuser,{TGT}Skdc`
`TGT{ClientId, Timestamp, Expir, SKdc }`
	The authentication response consists of two parts: (a) The user related information, which is encrypted with user's secret key and (b) the TGT which is encrypted with the KDC secret key and thus the user is not able to decrypt it. The user exploits the TGT in order to get a service ticket for accessing the requested service.

`(3){ServiceId, TimeStamp, Authenticator) {TGT} SKdc}`
`Authenticator = {ClientId, Timestamp } SKKdc`
	The user generates a ticket request that contains the authenticator, which is encrypted by the session key received in message (2), and the TGT.

`(4) {ClienId,TimeStamp,Expir,SKser,}SKKdc, {TKser}Kser`
`TKser {ClientId,ServiceId,Timestamp,Expir,SKser}`
	The KDC validates the request received in message (3) and generates a service ticket (TK$_{ser}$) that is encrypted with the service secret key and is sent back to the user along with the other information encrypted with service session key (SK$_{Kdc}$).

`(5) {ClientId,TimeStamp} SKser,`	In order for the client to access a service, he generates a

`{TKser}Kser`	request that contains his identifier and the timestamp encrypted with the service session key received in message (4). Along with this information he also sends the service ticket received in the message (4).
`(6) {Service response}`	This message is optional and is generated only if the client has requested to the service to prove its authenticity.

Table 1. Kerberos messages

Furthermore, Kerberos supports cross-realm operations between clients and service providers authenticated using ASs administered by different authorities. The administrators of the Kerberos realms must configure "inter-realm" keys to allow mutual authentication of entities from the different realms. Using cross-realm Kerberos supports the implementation of identity federations.

Cloud solutions can rely on Kerberos for authentication of the cloud infrastructure nodes and clients. For example, the HBase cloud database relies on Kerberos to authenticate the computing nodes [34] and the OpenStack cloud solution also supports Kerberos as a plugin in the keystone authentication service [35].

5.4 Identity Management in the Cloud Infrastructure as a Service

Though various open source platforms such as Nimbus, Eucalyptus, OpenStack, CloudStack and OpenNebula are available, OpenStack dominates the current market [36] for the deployment of Infrastructure as a Service (IaaS). OpenStack is a multi-stakeholder effort with broad participation (150+ companies) from some of the biggest IT vendors in the world including IBM, Dell, HP, Intel, AMD, Cisco, VMware, Yahoo! and AT&T, as well as Linux vendors Red Hat, SUSE and Canonical.

In the OpenStack platform the identity management, the authentication and the authorization services are provided by a specific service called **Keystone**. The identity service associates each user with a tenant and a role. In the context of OpenStack a tenant can be a project, a group or an organization that uses the cloud infrastructure. Under the "umbrella" of the tenants can be defined the users. The Keystone implements a role based access control system to control the access to the provided services. This approach is illustrated in Figure 7. Further, all the services employed in the cloud are configured with specific users assigning to them specific roles that are authenticated and authorized whenever it is required to provide a service as Figure 9 illustrates.

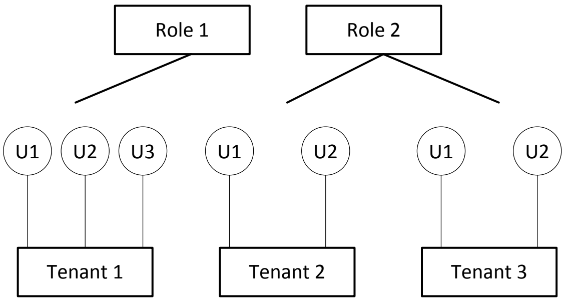


Figure 7. Keystone identity management architecture

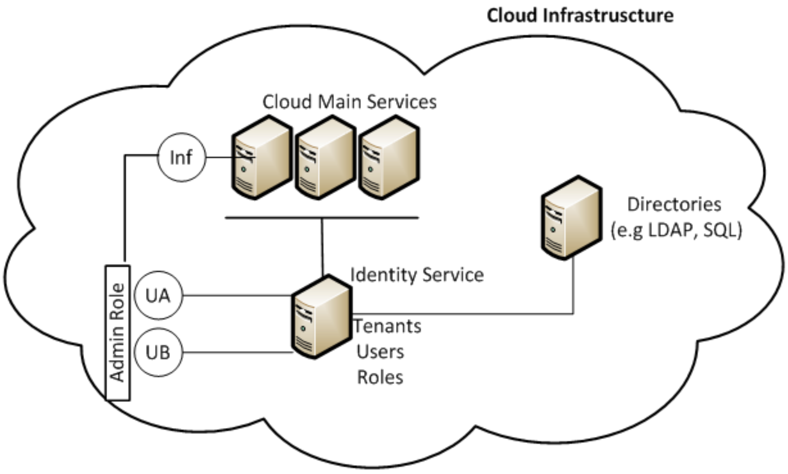


Figure 8. High level architecture for cloud identity management

6 Research Solutions

In this section we present a classification of existing identity management research solutions and a short description and reference to the main research papers of each class. Research solutions can be classified according to their focus considering the identity management architecture introduced in Section 5 in the following classes:

1. Complete models and architecture solutions for user-centric, single sign on, or federated identity management
2. Identity information model specifications using taxonomies and ontologies
3. Authentication solutions including biometrics (e.g. face recognition and fingerprint), context-based authentication, and combination of multiple authentication factors to increase trust
4. Policy-based solutions for specification and enforcement of authorizations and obligations to govern the access and usage of identity attributes (usage control)
5. Trust management with support to Levels of Assurance (LoA) and certificate chains analysis;
6. Approaches to mitigate privacy threats, privacy by design, attribute aggregation, and anonymization solutions
7. Security risks and threat analysis studies with respect to security and user privacy
8. Practical case studies

6.1 Identity Management Models and Architectures

Higgins [2] is an open source Personal Data Service (PDS) that allows user-centric management of identity attributes. The idea is similar to Windows Cardspace technology [3], where users have different identity cards with a pre-defined set of attributes that can be selected to be used with different service providers. The main component of the Higgins architecture is the Personal Data Service, which enables user-centric management of identity attributes and policies for each service provider.

In the EU project Hydra [4] an identity manager solution called Hydra Identity Manager (HIM) is proposed to comply with the ten laws of identity identified by the project based on a detailed analysis of security, trust, and privacy requirements in ambient environments. Their ten laws of identity are: (1) user empowerment: awareness and control, (2) minimal Information disclosure for a constrained use, (3) non-repudiation, (4) support for directional identity topologies, (5) universal identity bus, (6) provision of defining strength of identity, (7) decoupling identity management layer from application layer, (8) usability issue concerning identity selection and disclosure, (9) consistent experience across contexts, and (10) scalability. The authors claim HIM is the only to comply with their ten laws and show a detailed analysis and comparison of HIM in contrast to SAML, OpenID, Higgins, CardSpace, and Liberty Alliance approaches.

6.2 Identity Information Models

Layouni and Pollet [5] propose an ontological layer using the Web Ontology Language (OWL) to describe semantic models of the user, user interface, work domain, and information content of an identity federation. Their approach aims to define a standard vocabulary that is understood by all stakeholders. They demonstrate their approach for one scenario of identity federation but it is not clear how to apply their work for other scenarios. Their contribution is more on the use of ontologies for one identity management federation scenario and not on a general purpose ontology-based solution for any identity management scenario. A more general purpose ontology for identity credentials was also proposed by the National Institute of Standards and Technology (NIST).

6.3 Authentication

Agbinya et al. [11] proposes an authentication solution that combines face recognition and fingerprint authentication using an artificial neural network in order to increase the assurance level of the authentication process. The authors present the design and implementation of the solution, but no evaluation metrics with respect to performance and precision of the solution. A more comprehensive survey of this type of solutions has been conducted by the same authors [12] showing the strengths and weaknesses of different biometric authentication technologies and their fit for different application scenarios.

6.4 Usage Control: Authorizations and Obligations

With respect to usage control support, the PRIME project [7] proposes the only identity management solution that proposes to take into consideration the specification of obligations in addition to authorization policies with respect to the users' identity attributes. However, their research results and proposed XML policy schema language do not include the support for duties with temporal constraints. Obligations in their policy language simply consist of strings or lists of rights with respect to access to the identity attributes after released to third parties specifying recipients allowed, purposes allowed, actions allowed, and time durations for storage. It is not possible to specify duties with time duration constrains related to the storage of a specific identity attribute after a transaction or activity is completed. It is not possible to express with their policy language complex conditions with respect to activities for example, stating that the "credit card" identity attribute should be deleted just after an online purchase is completed, or that the "civil address" of a customer should not be stored after an item purchased is physically delivered.

The *PrivilEge and Role Management Infrastructure Standards* (PERMIS) [13] is a policy-based authorization solution that implements Role-Based Access Control (RBAC) using roles specified in X.509 certificates. PERMIS was originally developed to be used with the Shibboleth identity management system [14] but this is not mandatory, any other federated identity management solution could in theory adopt PERMIS. The main difference from standard RBAC is that PERMIS allows distributed assignment of roles since the X.509 certificates can be issued by multiple certificate authorities certified by an issuer up to a root self-signed certificate. PERMIS supports XACML for specification of authorization policies and SAML for exchange of attribute assertions. Since PERMIS relies in XACML, the support for specification of obligations is limited to strings usually without a precise meaning, or by means of standard propositional formulas [8].

6.5 Trust Management and Certificate Chains

Jøsang et al. [15] present a detailed analysis of the trust requirement in different identity management architectures. They analyzed the following architectures: isolated where each service provider manages the user identities independently, federated where credentials and identities are shared between providers, centralized where credentials only are shared between providers, and personal where the user manages the credentials and identities for each provider. The conclusion of their study is that user-centric personal identity management provides great flexibility with not many additional trust requirements in comparison with the other architectures, and seems the most promising approach.

Bhargav-Spantzel et al. [16] proposes to combine trust negotiation to better protect users' information when using identity federations. Their main motivation is the amount of management necessary from the user side if multiple identities and policies need to be defined for the combination of identity and service provider. Ideally, users should not be required to define policies individually for each combination, and trust negotiation could help to reduce the amount of work required from users. With trust negotiation, policies can be defined to allow any member of the trusted federation access to the users' identity attributes, without the need for individual policies.

6.6 Privacy by Design and Anonymity

One of the main anonymization solutions to support identity management systems is the Identity Mixer (Idemix) proposed by Camenisch et al. [17]. The idea behind Idemix is to use zero-knowledge proofs combined with encryption and signature schemes that allow users to reveal a selected part of their identity attributes. For example, a user may choose to assert the "city" attribute of their identity only without revealing their real name or detailed address information.

6.7 Risk, Threat, and Impact Analysis

Khattak et al. [19] and Gail-Joon & Sekar [20] propose approaches to perform risk and threat assessment of federated and user-centric identity management architectures. Lei and Takabi [21] show the security risks associated while using e-mail as online identity. In a nutshell, these approaches focus on risks for end users, relying parties, and service providers with respect to low levels of assurance of identities and privacy or identity theft risks for end users.

Alkassar and Husseiki [22] present an impact analysis of adoption Trusted Computing (TC) technology to support identity management in the context of the Fidis (Future of Identity in the Information Society) consortium. The use of TC has many opportunities with respect to assurance levels since the root of the authentication is a hardware secure tamper proof module, however, it is unclear how the privacy aspects of this adoption would impact end-users' privacy in real world deployments.

6.8 Practical Case Studies

White describes in [23] a case study of different identity management architectures in the Australian Public Sector. The author performed document analysis and interviews with designers of identity management architectures and identified the main patterns and differences in the designs to establish a common baseline. Common patterns are: information and data management, entity management, authentication, access control, provision management, credential management, directory services, meta-directories, governance, and privacy management. One interesting result of this study is the low concern of the interview participants with privacy in their architecture designs.

In [24] Rieger presents an hybrid approach that combines both user-centric and federated identity management that was developed to support the Max Planck Society using the SAML-based Shibboleth. This approach was used by the 80 autonomous institutes and relies on an identity proxy to provide interoperability between the different identity management technologies adopted in each institute. The selection of the respective home identity provider when a user tries to access a resource in a foreign institute is done based on the e-mail address of the user in a transparent way. The solution deployed in fact does not rely on user-centric identity management in the sense that the users are able to manage their own credentials. Details about the solution developed are also available in their website [25].

The STORK 2.0 project [26] proposes a set of pilot studies on the following application domains: banking, government, health, and learning. The studies are ongoing and there is no description of the results available yet.

7 Analysis of Stakeholders and Issues

Scope of this section to classify and analyze the stakeholders involved in an identity management process, while magnifying the related issues an identity management system should solve.

Figure 9 shows the main stakeholders in an identity provisioning scenario and the interactions between them. The Identity Owner authenticates with the Identity Provider and receives an authentication Token in case the authentication is successful. This Token is used by the Identity Owner to interact with a Relying Party, which verifies the Token with the Identity Provider and optionally retrieves identity attributes associated with this Token. The Identity Owner and the Relying Party are in many scenarios associated with a Service User and a Service Provider. This stakeholders and interactions may be to some extent different considering the technical solution adopted as described in Section 5.

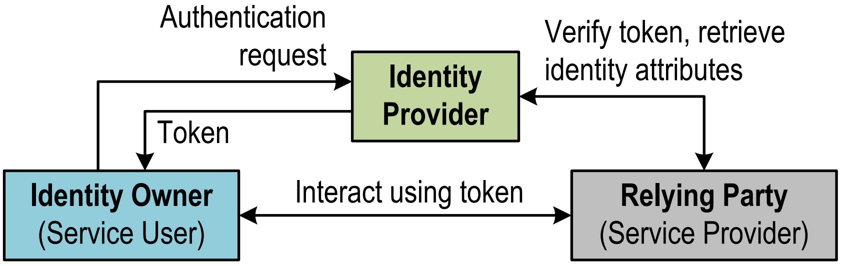

Authentication request
Identity Provider
Verify token, retrieve identity attributes
Token
Identity Owner (Service User)
Interact using token
Relying Party (Service Provider)

Figure 9 – Identity management stakeholder

Our objective in this section is to analyze the security requirements and issues for each stakeholder without considering the technical solution adopted.

7.1 Identity Owner

When authenticating with an identity provider, one important issue for the identity owner (a.k.a. service user) is the authentication method used to identify and verify the identity owner credentials. This authentication method may use one or many authentication factors including an username/password combination (mutually known secret), a PKI infrastructure with digital certificates, physical secure tokens (e.g. smart cards or hardware secure modules), biometric information, or context information that should not be easily tampered with (e.g. location assigned by a trustworthy sensor).

Before being able to authenticate, the identity owner must enroll with an identity provider, and decide upon the authentication credentials and methods supported. When enrolling with an identity provider, the identity owner is assigned a digital identity with attributes that may be certified up to a certain level of assurance (LoA). It is task of the identity provider to ensure the LoA is enforced and to securely store the authentication credentials. The secure storage of the authentication credentials is also expected from the identity owner, to prevent identity theft and impersonation. An identity owner may enroll with many identity providers with different LoAs and authentication factors, and therefore may possess a set of credentials that must be secure for his home environment, banking, or work environment.

An identity owner may face a series of security threats and risks, such as:

- Sharing of identity attributes between different identity providers allow re-identification and tracking of online activities. Shared identity attributes may imply that the user in fact is not as anonymous as expected;
- Low-level information collected by identity providers may allow more fine-grain tracking of identity and online activity, for example using network address (IP), WiFi connections, browser configuration, and cookies;
- Identity providers with low LoA and weak authentication process allows easy impersonation, for example, users can create as many digital identities with their own defined attributes without any verification during enrollment. This makes the digital identities less trusted;
- Identity attributes and credentials should be stored securely by identity providers and users;
- Identity owners should be able to specify access control policies to their identity attributes to be accessed by relying parties. Furthermore, policies should be defined to control rights and duties of relying parties after the identity attributes are accessed. Identity owners must be provided guarantees with respect to the enforcement of their policies;
- The specification and enforcement of the security policies with respect to identity attributes is also sensitive and difficult task. Identity owners should be supported in the specification of these policies and must be ensured that their policies are not publicly available.

A common practice is to provider software components to support end-users in the management of their online identities using desktop, mobile devices, or cloud services. Figure 10 shows an example of an End-user Identity Manager that manages the authentication tokens, credentials, and securely communicate this to the identity providers and relying parties.

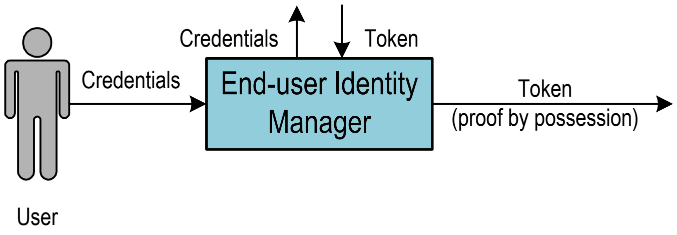


Figure 10 End-user identity manager

7.2 Relying Party

A relying party (a.k.a. service provider) is mostly concerned with the LoA of the identity attributes received and the trustworthiness of the digital identities. It is task of the relying party to securely store the identity attributes and session tokes received from the identity owners and to maintain and enforce a dataset of policies representing rights and duties with respect to the identity attributes.

An important aspect for the relying parties is the management of the trust relationships with users and identity providers. A possible approach is to observe the behavior of users with less trusted identities to detect anomalies that may be a result of identity theft and impersonation, such as access from unusual locations and time. Trust management tools may also be used to assess the reputation of identity providers from incidents published in the media.

7.3 Identity Provider

It is the identity provider duty to ensure secure authentication and session management between identity owners and relying parties. It is the identity provider task to select technical solutions that are proven secure and ensure the right implementation to prevent security incidents.

Furthermore, it should also manage and enforce usage control policies that control access and future usage of identity attributes by relying parties. Identity providers also need to manage trust relationships with other identity providers in case identity federations are supported, for example, to enable Single-Sign-On (SSO) between administrative domains.

The enforcement of rights and duties after relying parties retrieve identity attributes is a challenging task, and in case this cannot be achieved at least methods to ensure the detection of policy violations should be in place. One approach that could be used is the watermarking of identity attributes targeting particular relying parties, allowing future investigation in case leaks of identity attribute information is detected.

Identity providers are responsible for securely managing databases of an identity profiles, usage control policies, authentication sessions, and trust relationships with relying parties and other identity providers.

8 Privacy-Aware Multi-Morphed Digital Identities

As described in the previous sections, the electronic identity has a key role in enabling the citizen to access online services and digital information. On the other side it is potentially also a source of information leakage if not managed correctly. The situation is even more critical if we consider that several of the e-id used by the citizen to get access to personal information (e.g. facebook accounts, email accounts, online data repositories etc.) can be generally considered “soft-identities”, i.e. digital identities for which the registration and authentication mechanisms are generally based on weak mechanisms (an e-mail address is sufficient to obtain these identities and the authentication enforcement is generally based on the usual login-password credential).

The advent of smart-devices (e.g. smartphones) had the effect of exposing even more soft-identities to possible cyber-threats. To get integrated into the cloud and to provide the requested services, the smart-devices need to have access to the user’s e-IDs. In this case, it is not rare, for the end-user, to grant to the smart-devices a permanent access to his e-IDs (e.g. through the common function “*remind the credentials on this device*”). If on a side this allow to automatize several useful operations performed by the smart-device, on the other side, by delegating to it the authentication process, the end-user lose the full control on his own e-IDs and on the access to the services/data associated to it, dramatically jeopardizing their security.

In this section we define the set of requirements needed to design a framework to enable end-users to have full control on the information in possession of their smart-devices and more generally, on the information associated to their e-ids. The definition of a similar framework is on-going and what is described in this report is a first embryonic prototype aiming at studying new means to empower the control of the citizen on his personal data.

A similar framework should take in consideration three main concepts:

- **Digital-Identity**: i.e. a flexible mean to identify clearly actors interacting online and to define properties, features and rights of online services and information
- **Trust:** a sort of process allowing to automatically negotiate and establish the level of trust of a digital counterpart and associating consequently rights, permissions and obligations
- **Usage Control**: a mechanism to ensure that the data (or the right) obtained will be used only in the way defined by a certain policy.

From a functional point of view, the *privacy-aware-multi-morphed identity framework* should:

1. Enable the user to generate a set of customized digital identities, to be used in different context, and disclosing only the minimum portion of information required. It is important to remind here that with the term digital-identity we do not refer only to a digital certificate or to a set of credential, but to all the set of information (public or private) that, together concur at the characterization of the identity owner.

2. Allow, through a trust negotiation scheme, to define the level of trust of the digital counterpart, to identify the maximum amount of rights to be granted to the counterpart, and to disclose the minimum required amount of personal information to allow to the counterpart to establish, reversely, the level of trust required to access to a set of information/services
3. Define and enforce obligations on the data/services accessed by third parties, accordingly with their level of trust.

In the following of the chapter, the elements needed to implement this framework will be described.

8.1 Identity Metamodel

Figure 11 shows our identity metamodel. This metamodel is implemented in our prototype described in Section 9 and specifies the types of identity supported in an identity management system. Identity types can be physical or digital identities, and a special electronic document identity that is a physical type of one specific media (e.g. Smart Card) that contains a digital identity embedded on it. Digital identities are simply containers of named identity attributes verified using a verification method (e.g. face-to-face verification).

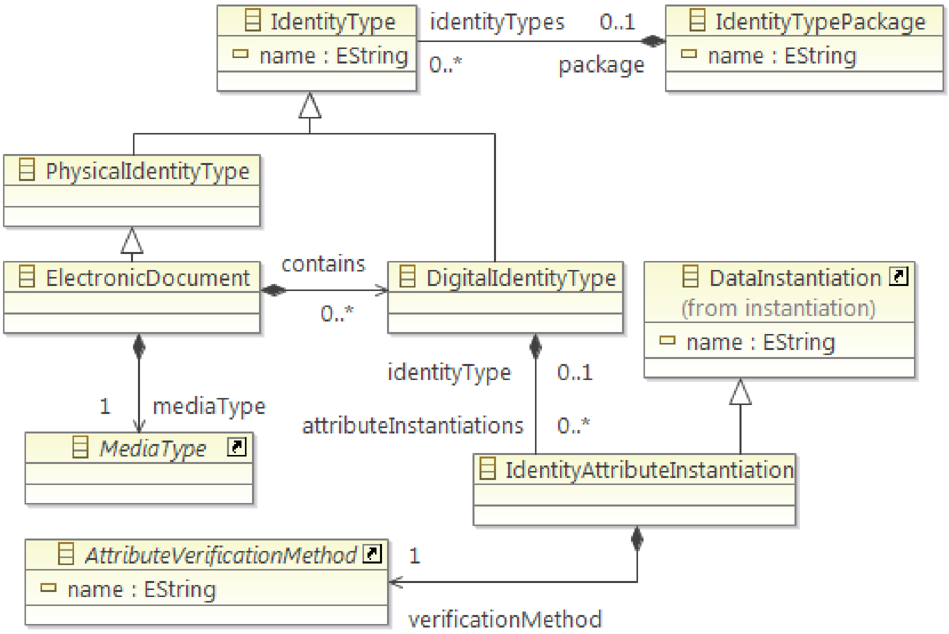


Figure 11 – Identity Metamodel

8.2 Multi-morphed identity, Inheritance and definition principles

Key element of our approach is the following consideration: when verifying our email, when connecting to online sharing services, when performing activities in our name, smart devices need to authenticate themselves. To do that, they use the soft-identities of their owner. In other words, smart-devices **inherit** the identity of their owner, i.e. the smart-device will be able to "impersonate" its owner while performing some operations. If, in a safe environment, this aspect can be considered acceptable, in an open and potentially adverse environment as the Internet, it might expose the citizen to a new set of threats against his privacy.

An additional aspect to take into consideration is related to the registration process of a new soft-identity. Let consider the following example: a user wants to get the access to an online forum. To do that, he need to pass through a registration phase, where he will need to provide a certain set of “sensitive” information, to get in change a couple id-password (his identity credential under a certain point of view), which will enable him to access to the forum.

The amount of personal information to provide varies from service to service. However, in several cases, due to the fact that it is impossible to establish a true trust level between the requester and the registration entity, what is declared does not provide any guarantee of truth, leaving the service-providers completely unprotected against malicious users willing to obtain false identities to be used for more complex illicit operations.

To cope with the presented problems, digital citizens should be able to generate soft-identities guaranteeing, on a side, the minimum possible data-disclosure, and on the other, when required, providing means to the service-provider to get an acceptable level of assurance on the reliability of the information provided, without impacting on the right of privacy of the end-user.

In our approach, a soft electronic identity (e-id) (intended not only as a credential, but also as all the set of information produced to build it) can contain:

- **Uncertified and anonymous information**, i.e. information freely provided by the owner without any guarantee of truth
- **Signed un-certified information**, i.e. information freely provided by the owner of the device and signed with his private key to guarantee that this information has been indeed provided by him, but without any guarantee on the truth of what claimed
- **Certified information**, i.e. information directly coming from a portion of the owner's hard e-id or certified by a trusted third party

It is evident how, this information classification, directly implies different level of trust.

It is not intention of this study to describe technically how a soft-id can be created on the basis of these information, as this will be explained in a further, specific report.

However, we anticipate here in form of list, some of the techniques which will be explored to combine together these three classes of information to protect on a side the privacy of the citizen while guaranteeing at the same time an acceptable level of trust with respect to the criticality of the service needed:

- Reputation mechanisms to verify the level of trust when using signed un-certified information
- Evidence-graphs, to combine uncertified and anonymous information to verify their level of consistency
- Blind signature schemes and zero-knowledge proofs with regard to “certified information”, to protect the privacy of the citizen and at the same time leverage on crypto mechanisms to obtain the higher level of trust

On the basis of the kind of information constituting a soft e-id (anonymous information, signed, certified) it would be possible to establish some trust rank.

8.3 Trust Negotiation

A Trust negotiation is an interactive process between two parties having the goal of establishing mutual trust to release a given resource or a part of it, considering the trust level achieved. In our case, a similar negotiation is needed to allow entities, such as smart-devices, rescue authorities, and end-users, to automatically identify the level of trust of the counterpart. After the level of trust is identified, they can decide if a certain request, either for information or action should be granted or partially allowed.

There exist several trust negotiation schemes in literature [38][39][40], to our purposes we adopted the Trust-X[41] schema as it appears to be to our knowledge the most suitable for our needs:

- It was designed on the principle that two parties can establish trust directly without involving trusted third parties, other than credential issuers
- It was specifically designed for a P2P environment
- Each party can alternatively act as requester or resource controller in different negotiations
- It is supported by a XML-based Resource Negotiation Language, **X-RNL** allowing to specify negotiation policies over resources of any type
- It allows to express group-based policies in which resources belong to more than one peer
- It provides an interpretation of the language formulae in terms of states of peers

For further details on an extension of Trust-X toward critical infrastructure protection please refer to [42].

8.4 Usage Control

Usage control extends access control with the concept of obligations, which specifies constrains on the use of data after access is granted. In our solution we apply an usage control framework previously developed by us [27] that supports the specification of policies using mechanisms according to an Event-Condition-Action (ECA) rule structure. These mechanisms have their **Action** part executed when an **Event** pattern is observed and the **Condition** expression evaluates to true.

In order to support both preventive and detective enforcement two types of events are specified: tentative and actual events. Tentative events indicate that an activity (action or interaction) is ready to be started in the monitored system but has not yet started, and actual events indicate that an activity has been completed.

For tentative events the Action part of a mechanism specifies an authorization action that allows the activity to be executed, deny the execution of the activity, delay the execution, or modifies the execution. For actual events the Action part simply triggers the execution of additional actions.

The Condition expression of a mechanism supports propositional, temporal, cardinality, and event operators. Our language for specification of conditions is based on Linear Temporal Logic (LTL) and evaluates event traces considering a discrete timestep with a predefined granularity. The condition part of a mechanism can be parameterized with variables, which allows re-use and modularity in the specification of usage control policies.

From a usage control perspective, our mechanisms can be used to specify authorization and obligations without any changes. Authorizations are essentially mechanisms specifies by domain administrators to be enforced on their own domains. Obligations are mechanisms specified by domain administrators that are delegated to other domains when interactions that exchange sensitive data take place.

In contrast to existing frameworks for access and usage control our usage control framework is more expressive and can express complex authorizations and obligations. For example, using existing access control languages such as XACML [8] a policy stating that access should be denied to users after three unsuccessful logins cannot be expressed because XACML does not support cardinality operators.

Section 9 presents our ongoing prototype implementation that supports the specification and enforcement of usage control policies. Our prototype only describes the specification support since the enforcement support is not yet integrated in our testbed described in Section 10.

8.5 Architectural Design

In this section we provide a functional view of our Trusted Usage Framework (TUF) using the building blocks described in the previous subsections. We identify two phases in the operation of the framework: a **setup phase** in which the owner of an hard e-id builds using a soft e-id generator a set of soft e-ids which will be left as inheritance to the smart-devices owned by the user, and an **operational phase** in which the smart-device is called to interact with the external world and provide information/services according to a set of policies.

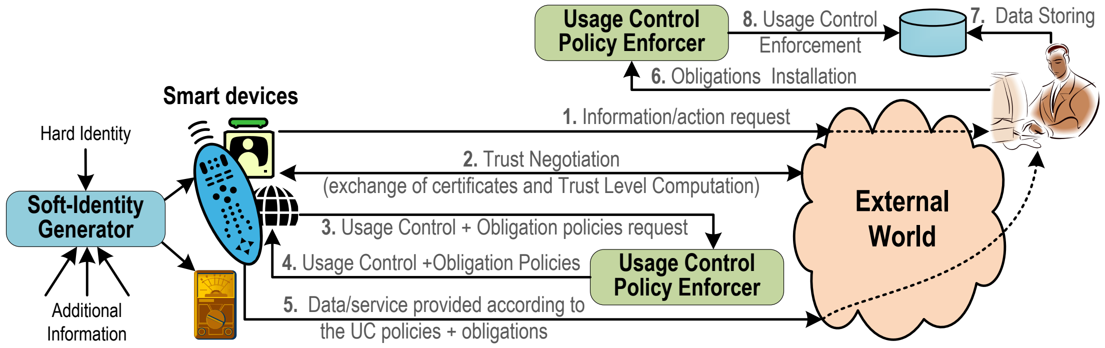

Usage Control Policy Enforcer
8. Usage Control Enforcement
7. Data Storing
6. Obligations Installation
Smart devices
Hard Identity
Soft-Identity Generator
Additional Information
1. Information/action request
2. Trust Negotiation (exchange of certificates and Trust Level Computation)
3. Usage Control + Obligation policies request
4. Usage Control +Obligation Policies
Usage Control Policy Enforcer
5. Data/service provided according to the UC policies + obligations
External World

Figure 12 Framework Functional Overview

More in details, this last phase is articulated as follows:

- An external entity requests access to information or to execute an operation by the smart-device
- The smart device, prior to provide what requested, engages with the external entity a trust negotiation. The trust negotiation is bilateral, i.e. both the parties will be called to provide proof of their level of trust

- Once the trust negotiation is finalized, a level of trust associated to the external entity is established. The smart-device then interrogates the **Usage Control Policy Enforcer** (UCPE) asking for instructions on what to provide to the external party on the basis of the level of trust established
- The UCPE provides the corresponding disclosure/execution policy and a set of obligations that must be fulfilled by the external party in order to obtain the data
- The smart device sends the data allowed according to the policies, executes the allowed actions, and sends a set of obligations to the external party
- The external party enforces the obligations
- The received data is stored
- The external-party's UCPE monitors the usage of the data obtained respecting the new obligation received

It is important to notice that the information is provided to the requesting party only when the requester's UCPE would provide acknowledgment on the installation of the obligations.

9 Prototype Implementation

In this section we present our prototype implementation that supports the specification of authorizations and obligations as Event-Condition-Action (ECA) enforcement rules. These rules use as a reference a set of inter-related design models representing different aspects of the identity management system, and are used as input for the runtime components in the framework. This solution to enable monitoring of ECA rules and execution of security enforcement behavior is named the Model-based Security Toolkit, or just SecKit [27],[28].

The SecKit consists of a collection of metamodels for specification of a computer system structure, information, behavior, context, identities, organizational roles, and security rules. These metamodels provide the foundation for security engineering tooling add-ons and metamodel extensions to address requirements of governance, security and privacy. Figure 13 gives a high-level overview of the design models supported by the SecKit, which are: system (structure, information, behavior), context, identity, role, and security rules. These models provide the foundation for the design and runtime tooling, and extensions/add-ons focusing on specific security aspects of a computer system.

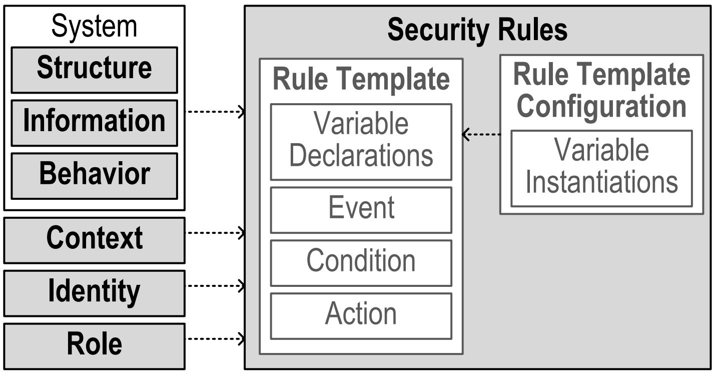


Figure 13 – Design models supported by the SecKit

The first step using the SecKit is the specification of the System behavior, structure, and information model. The SecKit adopts a generic design language to represent the architecture of a distributed system across application domains and levels of abstraction including refinement relations support inspired in the Interaction System Design Language (ISDL) [29].

Figure 14 shows the behavior design model of the identity management scenario proposed in this report. The identity owner creates a hard identity and specifies policies governing the access to the hard identity attributes. The identity owner uses this hard identity to create a soft identity, and also specifies policies to govern the access to the soft identity attributes when accessing a service provided by a relying party the identity owner uses a set of trust negotiation rules to decide the policies for the soft identity being used in the access. The relying party is then able to retrieve the soft identity attributes and should enforce the policies provided by the identity owner. The details about this interaction and the information exchanged are depicted in the behavior model.

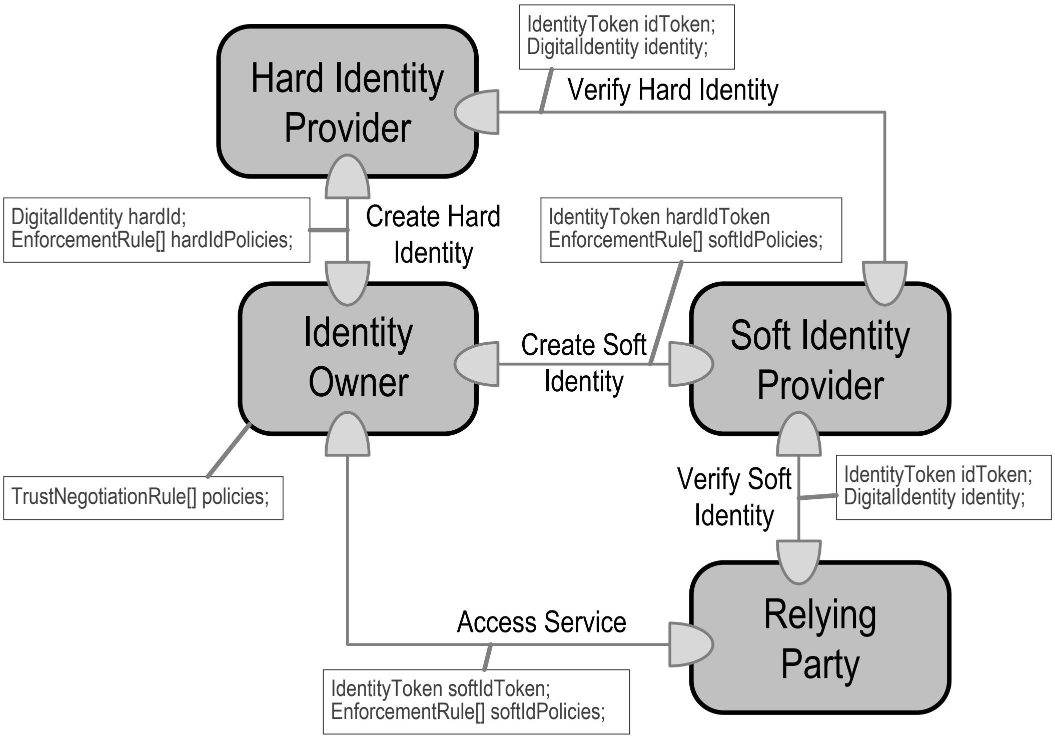

IdentityToken idToken;
DigitalIdentity identity;
Verify Hard Identity
Hard Identity Provider
DigitalIdentity hardId;
EnforcementRule[] hardIdPolicies;
Create Hard Identity
IdentityToken hardIdToken
EnforcementRule[] softIdPolicies;
Identity Owner
Create Soft Identity
Soft Identity Provider
TrustNegotiationRule[] policies;
Verify Soft Identity
IdentityToken idToken;
DigitalIdentity identity;
Access Service
Relying Party
IdentityToken softIdToken;
EnforcementRule[] softIdPolicies;

Figure 14 – Identity management behaviors

In addition to the specification of the system models the SecKit also includes metamodels for specification of Context, Identity, and Role models. The identity model specifies the identity types and attributes that are allowed in this identity types. The role model specifies the organization role hierarchy, with the possible of inheritance of membership. For example, *Doctor* and *Nurse* could be specified as sub-roles of the *Health Professional* role.

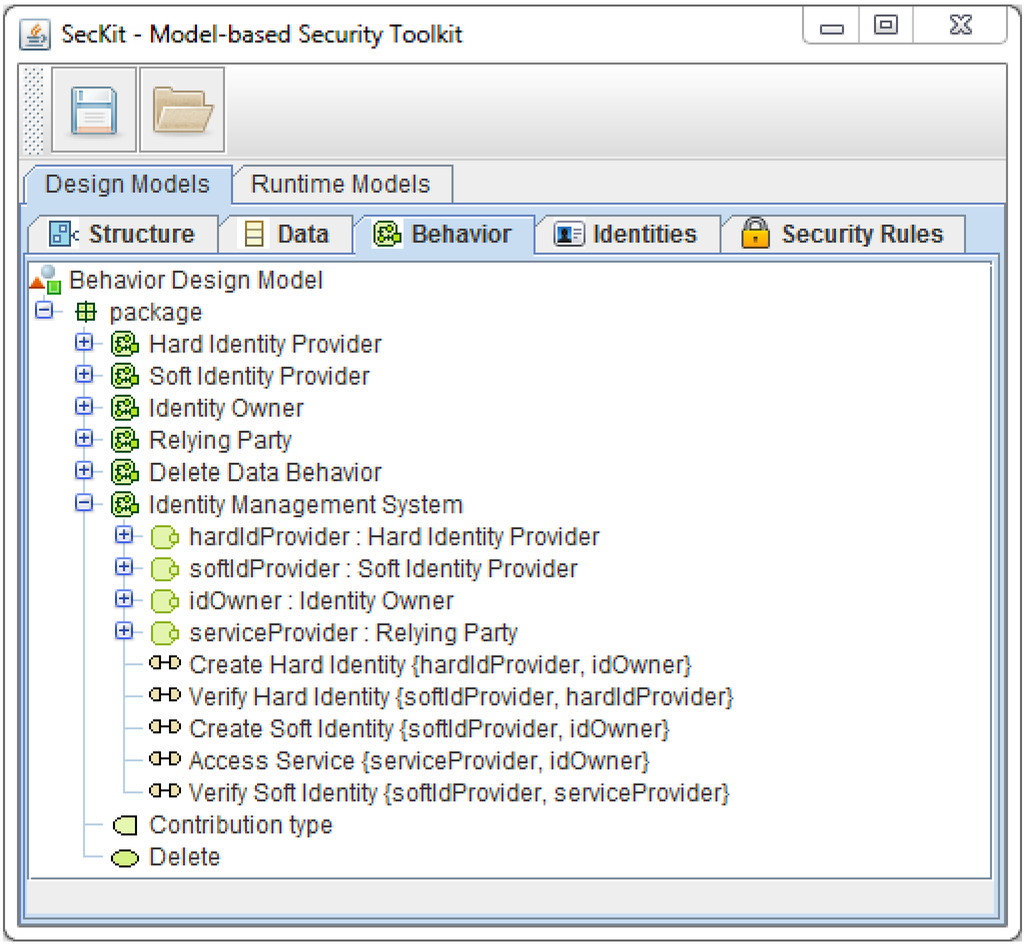
The context model specified types of Context Information and Context Situations. Context Information is a simple type of information about an entity that is acquired at a particular moment in time, and Context Situations are a complex type that models a specific condition that begins and finishes at specific moments in time. For example, the *GPS location* is an example of a context Information type, while *Fever* and *In One Kilometer Range* are examples of situations where a *patient* has a temperature above 37 degrees Celsius and a *target* entity has a set of nearby entities not further than one kilometer away. *Patient* and *target* are the roles of the different entities in that specific situation.

The specification of authorization and obligation policies is done in the SecKit using an Enforcement Rule model containing *Rule Templates* that must be explicitly instantiated using *Rule Template Configurations*. A rule template follows an ECA semantics defined over discrete traces of sets of events, when the trigger event (E) is observed and the condition (C) evaluates to true the action (A) is executed. Templates are parameterized with variables that are instantiated by the template configuration. The Rules specified using the SecKit make reference to the design models of the system (structure, behaviour and information), roles, context, and identities.

We model the start of an activity, ongoing activities, and the completion of an activity with the event modalities: *start*, *ongoing*, and *completed*. To support enforcement of usage control policies including authorization decisions we model *tentative* and *actual* events. A tentative event is generated when an activity is ready to be started but has not yet started, giving the opportunity for the execution of enforcement actions.

A tentative event may trigger the execution of an enforcement behavior to allow or deny the execution of the activity. If the activity is allowed it is also possible to specify an optional modification or delay of the activity execution, for example, anonymizing activity data before the activity takes place. The execution part of an enforcement template may trigger the execution of additional activities, for example, notifications or logging of information.

Figure 15 shows the same behavior model depicted in Figure 14 now specified using our prototype implementation. Our prototype offers support for specifying the different design models supported by the SecKit and also runtime enforcement models to support behavior execution monitoring and enforcement of the security rules.



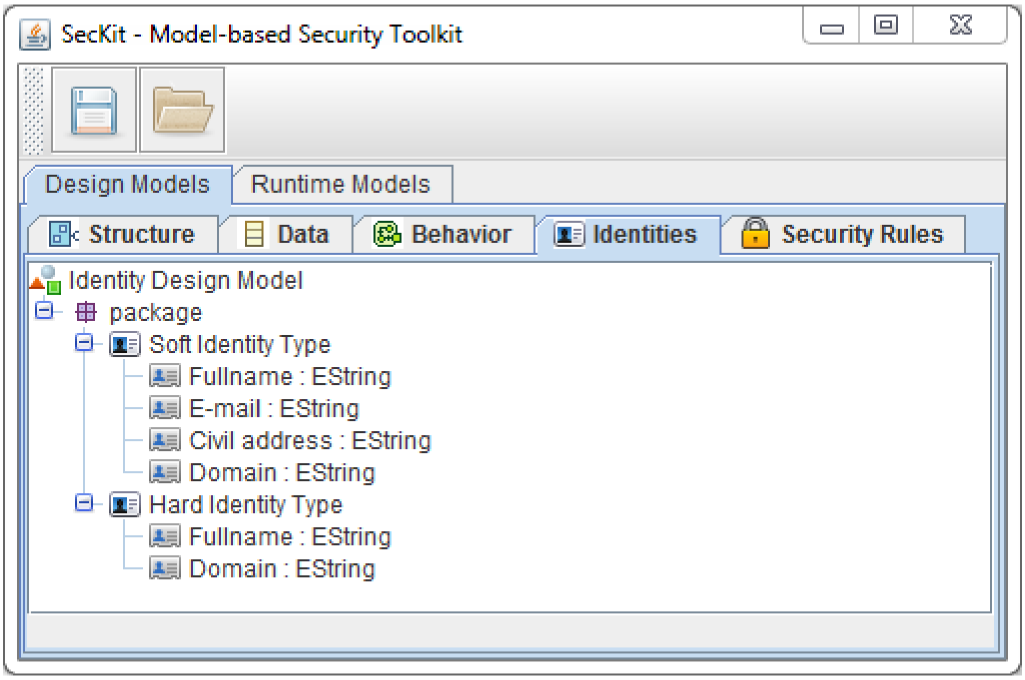
- Behavior Design Model
  - package
    - Hard Identity Provider
    - Soft Identity Provider
    - Identity Owner
    - Relying Party
    - Delete Data Behavior
    - Identity Management System
      - hardIdProvider : Hard Identity Provider
      - softIdProvider : Soft Identity Provider
      - idOwner : Identity Owner
      - serviceProvider : Relying Party
      - Create Hard Identity {hardIdProvider, idOwner}
      - Verify Hard Identity {softIdProvider, hardIdProvider}
      - Create Soft Identity {softIdProvider, idOwner}
      - Access Service {serviceProvider, idOwner}
      - Verify Soft Identity {softIdProvider, serviceProvider}
    - Contribution type
    - Delete

Figure 15 – Model of identity management system behaviour with interactions

Each interaction depicted requires a level of trust that is negotiated during the initialization phase of the interaction. The established level of trust after the negotiation authorizes the successful completion of the interaction, which possibly implies the execution of an operation and/or exchange of sensitive data. In order to govern the authorization to execute the operation or access to the sensitive data we apply the runtime models and enforcement components provided by the SecKit.

Furthermore, we exchange with the interacting parties obligation policies that regulate how the data exchanged should be used. Trust levels, their corresponding authorization policies, and obligations must be specified for each interaction considering their impact and sensitivity.

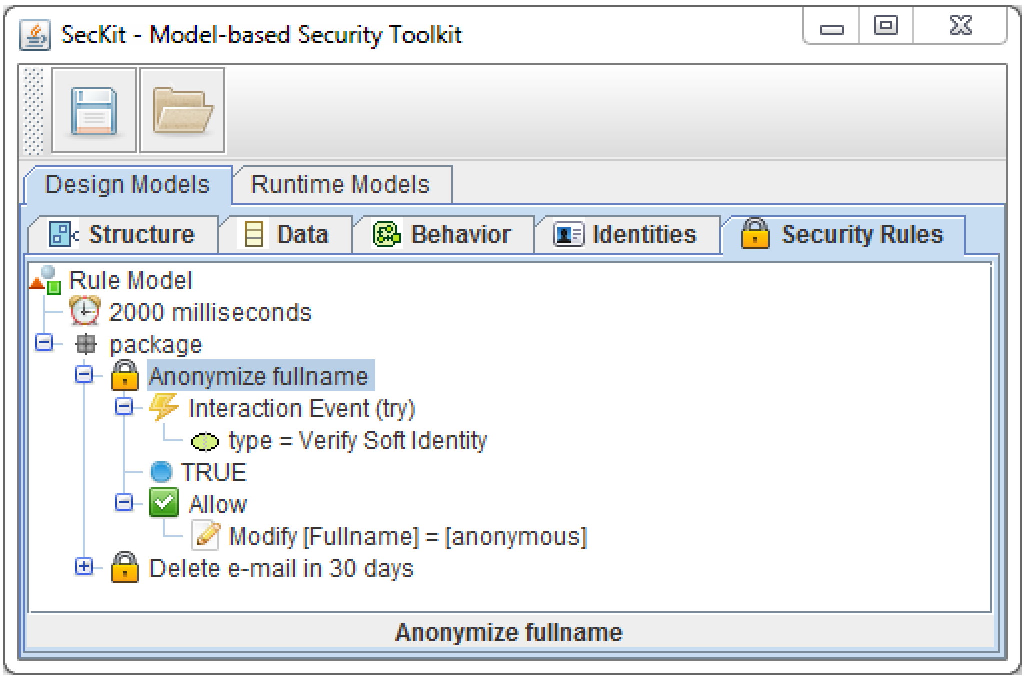
Figure 16 shows an identity model example specified using the SecKit prototype. In this figure we show two identity types representing the concept of hard and soft identities.



Identity Design Model
- package
  - Soft Identity Type
    - Fullname : EString
    - E-mail : EString
    - Civil address : EString
    - Domain : EString
  - Hard Identity Type
    - Fullname : EString
    - Domain : EString

Figure 16 – Model of soft and hard identity types with attributes

Figure 17 shows one security rule template that anonymizes the *Fullname* when the “Verify Soft Identity” interaction is about to happen (try). The anonymization in this case is simply a replacement of the *Fullname* attribute value to the string “anonymized”.

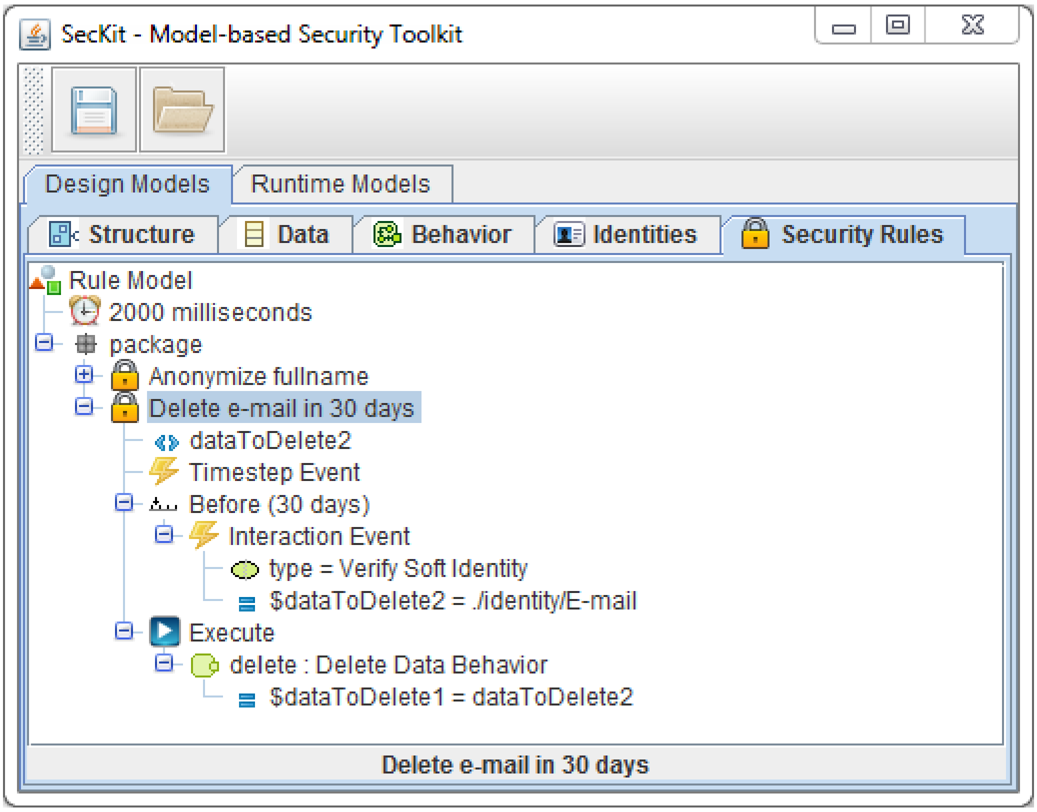


- Rule Model
  - 2000 milliseconds
  - package
    - Anonymize fullname
      - Interaction Event (try)
        - type = Verify Soft Identity
      - TRUE
      - Allow
        - Modify [Fullname] = [anonymous]
    - Delete e-mail in 30 days

Anonymize fullname

Figure 17 – Policy to anonymized soft identity before the verification

Figure 18 shows an addition more complex security rule template that is trigger 30 days after the “Verify Soft Identity” interaction is executed. This rule triggers the execution of the “Delete Data Behavior”, which deletes the E-mail identity attribute part of the verified identity. The specification of behavior triggered by the security rules is done using the standard support for behavior specification.



- Rule Model
  - 2000 milliseconds
  - package
    - Anonymize fullname
    - Delete e-mail in 30 days
      - dataToDelete2
      - Timestep Event
      - Before (30 days)
        - Interaction Event
          - type = Verify Soft Identity
          - $dataToDelete2 = ./identity/E-mail
      - Execute
        - delete : Delete Data Behavior
          - $dataToDelete1 = dataToDelete2

Delete e-mail in 30 days

Figure 18 – Policy template to delete e-mail identity attribute 30 days after soft identity is verified

Figure 19 shows the specification of the “Delete Data Behavior”, which is parameterized with one variable specifying the data to be deleted. When this behaviour is instantiated in the security rules this variable must be also assigned a value.

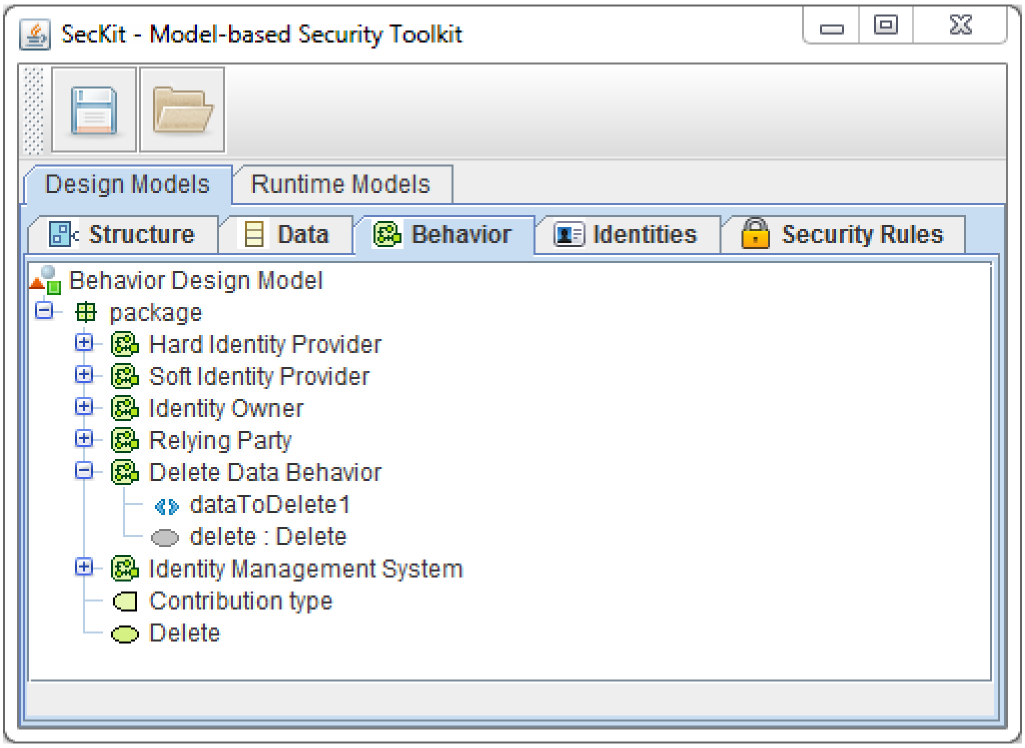


Figure 19 – Delete data behaviour type parameterized with variable

Behavior instantiations executed by security rules can also be assigned a target entity, or in case this target entity is not specified the rule engine itself instantiates the respective behavior. The target entity must support runtime extension with security rule behaviors in order to allow the execution of the security rule behavior instantiation.

Using the SecKit enforcement rules, policies can be specified for authorizations and obligations inside an outside of an administrative domain. For example, the identity owner can specify the instantiation of an enforcement rule that should be evaluated by a service provider. The delegation of policies and mutual establishment of domain identities is done using a trust negotiation approach.

10 Cloud Testing Facility

To analyze the impact of soft-identities solutions on modern online services a testbed allowing to simulate real-online user experiences is needed.

As the majority of modern online services is based on the raising cloud paradigm, we designed and deployed within the experimental open-space of the Digital Citizen Unit an infrastructure able to reproduce in a flexible way cloud and mobile-cloud use case scenarios.

This section briefly provides technical details on such infrastructure.

Though various open source platforms such as Nimbus, Eucalyptus, OpenStack, CloudStack and OpenNebula are available, OpenStack dominates the current market [36] for the deployment of Infrastructure as a Service (Iaas). OpenStack is a multi-stakeholder effort with broad participation (150+ companies) from some of the biggest IT vendors in the world including IBM, Dell, HP, Intel, AMD, Cisco, VMware, Yahoo! and AT&T, as well as Linux vendors Red Hat, SUSE and Canonical.

Thus, we rely on OpenStack to deploy a cloud infrastructure as a service that will be used to understand the threats introduced by the cloud against digital identities schemes and research the possibilities of implement solutions for enhancing the digital citizens' trust towards the provided services.

10.1 OpenStack Services

The OpenStack platform, as depicted in Figure 20, is composed of different services that might be hosted on the same or multiple nodes depending on the architecture.

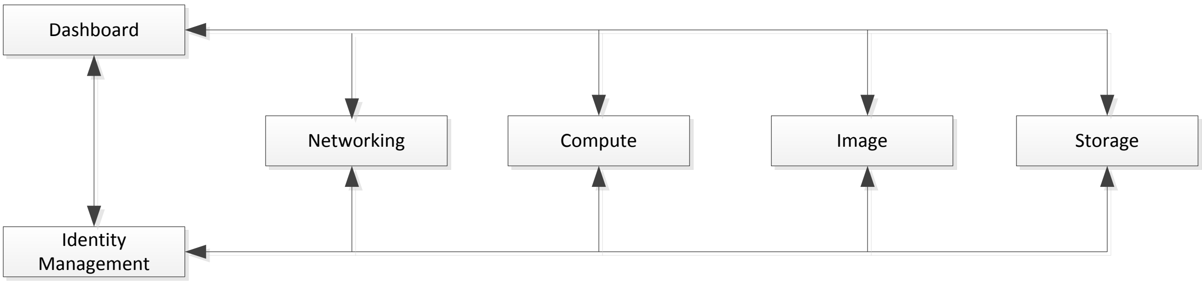


Figure 20. OpenStack services communication interactions

In the following subsections we briefly describe these services and the minimum hardware requirements for employing a private cloud infrastructure.

10.1.1 Nova Compute Service

The Nova Compute is the core service of the OpenStack framework providing the virtualization layer between the operating system and hardware, which enables the management of virtual machines.

10.1.2 Image Service

The image service manages the images stored in the cloud that users can instantiate latter. Image administration is usually restricted to specific users. This approach should be followed in order to eliminate the chances of uploading a malicious Image to the cloud infrastructure.

10.1.3 Storage Service

OpenStack supports object and block storages through swift and cinder services correspondingly. The object storage (swift service) is suitable for providing storage on “static” data (e.g., images) through HTTP. The swift service is not relied on centralized controller providing this way object storage scalability.

The block storage (cinder service) provides volumes of storages as individual hard drives – where each of these volumes is managed individually. The block storages is the basis of storage area network services in the cloud. It should be noted that since each volume is attached to an individual virtual server- as a result the block storage cannot be used for sharing data among different virtual machines.

10.1.4 Virtual Networking (Quantum)

The Quantum service provides the appropriate features for networking virtual machines both for internal and external communication. This is achieved by providing virtual networking components such as routers, networks, and other related services.

10.1.5 Dashboard

The Dashboard is the front-end service providing a user friendly interface (through web) to manage the cloud infrastructure. The Dashboard provides functionalities such as virtual machine creation and revocation, as well as virtual network management.

10.1.6 Identity Management

The OpenStak Identity service authenticates and authorizes access to the cloud resources for all possible entities that exist in the cloud infrastructure. This means that all the services running inside the cloud should be authenticated and authorized towards the identity management services. For more details please refer to Section 5.

10.1.7 Hardware requirements

The following table specifies the minimum hardware requirements as defined in [44].

	Controller	Network	Compute
Minimum num. of disks	2	1	1
External + API network	Yes	Yes	No
Management network	Yes	Yes	No
VM Data network	No	Yes	Yes
Num. of networks	2	3	2

Table 2. Architecture and node information

10.2 G7 Cloud Infrastructure as a Service

In this section we provide a brief description of the configuration of the cloud infrastructure implemented in the Digital Citizen Security laboratory.

10.2.1 Configuration

G7 cloud architecture relies on the OpenStack platform and consists of the following components:

- **Controller Node:** Provides all the appropriate interfaces and services for managing the cloud infrastructure. For instance, hosts a web service allowing users to initiate their virtual machines and manages the lifecycle of virtual machines.
- **Network Node:** Manages virtual networks and provides the necessary network services such as DHCP, Layer 3 Routing, Layer 2 switching and IP addressing.
- **Compute Node(s):** Host the virtual machines.

The high level architecture of G7 cloud infrastructure is illustrated in the Figure 21. We rely on this this setup because it can be extended easily for achieving high availability. It should be noted that the current deployment does not support high availability.

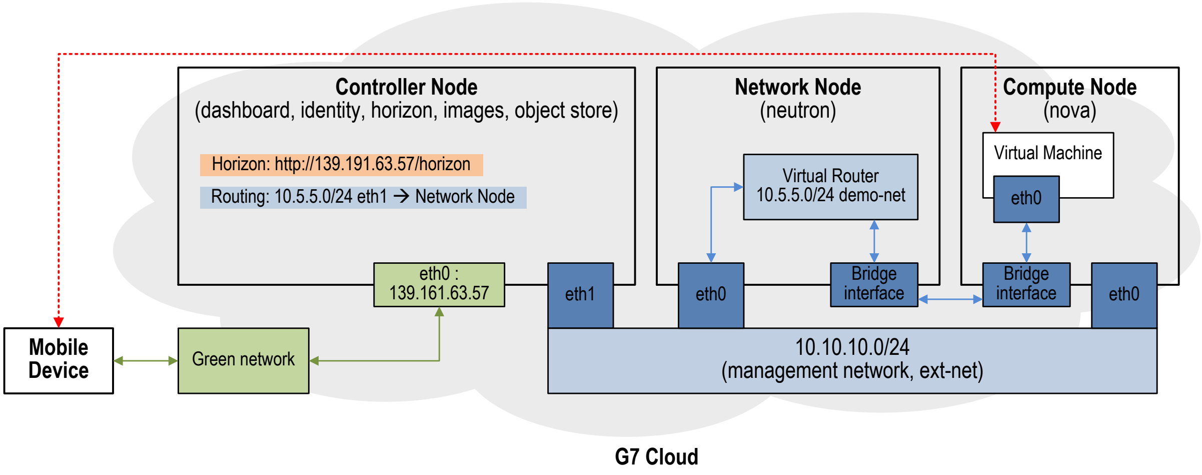


Figure 21. G7 OpenStack Cloud Architecture

Component	Services	Network Interfaces	Memory	Hard Disk
Controller	Quantum-Server Nova-API Nova-Scheduler Keystone RabittMQ MySQL	2	4 GB	2

Network	Open vSwitch Quantum-plugin-openvswitch-agent Quantum-metadata-agent Quantum-l3-agent Quantum-dhcp-agent	2	4 GB	1
Compute	Nova-compute-kvm Open vSwitch Quantum-plugin-openvswitch-agent	2	4 GB	1

Table 3. Deployed services and hardware configuration for G7 cloud architecture

10.2.2 Networking

As illustrated in the Figure 21, the following three different networks have been deployed in order to support the G7 cloud infrastructure:

- **Management network:** Used for management communication among the cloud nodes.
- **VM Network:** Enables the communications for VM with the cloud (infrastructure) services whenever needed.
- **Public Network:** Provides connectivity to VMs to other public networks such as Internet.

10.2.3 Problems and Solutions

During the deployment of G7 cloud infrastructure we face various issues, which are summarized in Table 3.

Service	Affected Node(s)	Problem	Solution
openvswtich-switch	Compute/Network	The service was not stated. “Error [quantum_aegent] Failed to create ovs patch port. Cannot have tunneling enabled for this agent”. This is because the installed package does not support tunnels.	1. install –y openvswitch-datapath-sources 2. install –y module-assistant 3. module assistant prepare 4. module-assistant auto-install openvswitch-datapath 5. In case that module does not found your linux version do: a. uname –r (this gives linux version) b. cd /lib/modules/(linux –veresion)/build/include/linux c. ln –s –o /generated/uapi/linux/version.h 6. re-run command four
	Compute	VM was not possible to start from CLI. Error: nova.compute.manager attribute error object has no attribute	When you create a new virtual machine you have always to connect it to a network.
Network	Virtual Machines	Udhcp started. Sending discover. In some cases the dnsmasq service does not return the IP address to the virtual machines.	1. Delete the last record from the dns (var/lib/quantum/dhchp/some-number/host) 2. kill –s HUP id of dnsmasq service

Table 4. Problems and indicative solutions for cloud deployment

11 Conclusions

The main contributions of this report are:

- A preliminary analysis of the technical and scientific literature on digital identity management schemes
- A model describing the interactions of the actors involved in the digital identity management process
- The description of the main elements composing a framework to empower the control of the citizen on his own persona digital information
- The deployment of a first working prototype applying the usage-control paradigm on the management of soft-digital identities
- The deployment of an experimental platform for privacy and security test in the cloud.

On the light of this initial study, it is evident how the pervasiveness of the digital world is quickly changing the life style of the citizen, which is relying more and more on digital-services to perform every-day operations. The phenomenon has recently received an additional burst thanks to the increasing diffusion of smart-devices and the consequent increasing inclusion of the citizen in the digital world. While on a side such a technological evolution makes finally possible the deployment of the so called Smart-Homes & Smart Cities in an Internet of Things fashion, on the other side, citizen privacy and security are becoming more and more exposed to cyber-threats. Smart-devices are becoming the repositories of a huge amount of information related to the personal life of the citizen and, in some cases they might have a critical role when called to perform actions impacting on the citizen's life. Key point of this digital revolution is the online-identity of the citizen, intended as the set of mechanisms allowing to the citizen and to digital services to recognize each-other and on the basis of that, to establish a chain of trust, rights and permissions.

In this context, while a clear direction have been taken when considering the so called hard e-id (e.g. digital identities issued by governments under strict registration rules and management constraints), standardizations and regulations are still missing for what concern soft-id. The impact of this weakness cannot be considered negligible, as the majority of the daily online operations performed by the citizen rely on soft e-id and not on hard e-id.

The protection of the citizen’s online privacy is a complex problem that needs to be tackled from multiple sides. Under this light, there is a strong need for mechanisms allowing to create a chain of trust between online actors, mechanisms to regulate the online information flow, to guarantee the citizen’s anonymity while at the same time providing assurance to the service provider on the trust level of the end-user. All these mechanisms will have to be coherently linked together under the umbrella of digital identity of citizens and smart-devices.

As explained in the introduction this contribution constitutes only the first step of a more extensive research activity which aim at exploring the aspects of digital privacy and anonymity of the online citizen.

References

[1] C. Neuman, T. Yu, S. Hartman, K. Raeburn. The Kerberos Network Authentication Service (V5). IETF RFC 4120. Available at: http://tools.ietf.org/html/rfc4120.

[2] Higgins Personal Data Service. Available at: http://www.eclipse.org/higgins.

[3] David Chappell. Introducing Windows CardSpace. Available at: http://msdn.microsoft.com/en-us/library/aa480189.aspx.

[4] Akram, H.; Hoffmann, M., "Supports for Identity Management in Ambient Environments - The Hydra Approach," *Systems and Networks Communications, 2008. ICSNC '08. 3rd International Conference on* , vol., no., pp.371,377, 26-31 Oct. 2008, doi: 10.1109/ICSNC.2008.77

[5] Layouni, F.; Pollet, Y., "An Ontology-Based Architecture for Federated Identity Management," *Advanced Information Networking and Applications, 2009. AINA '09. International Conference on* , vol., no., pp.162,166, 26-29 May 2009, doi: 10.1109/AINA.2009.124.

[6] NIST DRAFT SP 800-103. An Ontology of Identity Credentials, Part I: Background and Formulation. Available at: http://csrc.nist.gov/publications/PubsDrafts.html.

[7] PRIME Project. PRIME - Privacy and Identity Management for Europe. Available at: https://www.prime-project.eu.

[8] Erik Rissanen. eXtensible Access Control Markup Language (XACML) Version 3.0, OASIS Standard, Jan-2013, Available at: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf.

[9] Bhargav-Spantzel, A.; Squicciarini, A.C.; Bertino, E., "Trust Negotiation in Identity Management," *Security & Privacy, IEEE* , vol.5, no.2, pp.55,63, March-April 2007, doi: 10.1109/MSP.2007.46;

[10] Elahi, G.; Lieber, Z.; Yu, E., "Trade-off Analysis of Identity Management Systems with an Untrusted Identity Provider," *Computer Software and Applications, 2008. COMPSAC '08. 32nd Annual IEEE International* , vol., no., pp.661,666, July 28 2008-Aug. 1 2008, .doi: 10.1109/COMPSAC.2008.164;

[11] Agbinya, J.I.; Mastali, N.; Islam, R.; Phiri, J., "Design and implementation of multimodal digital identity management system using fingerprint matching and face recognition," *Broadband and Biomedical Communications (IB2Com), 2011 6th International Conference on* , vol., no., pp.272,278, 21-24 Nov. 2011.

[12] Mastali, N.; Agbinya, J.I., "Authentication of subjects and devices using biometrics and identity management systems for persuasive mobile computing: A survey paper," *Broadband and Biomedical Communications (IB2Com), 2010 Fifth International Conference on* , vol., no., pp.1,6, 15-17 Dec. 2010
doi: 10.1109/IB2COM.2010.5723618

[13] *PrivilEge and Role Management Infrastructure Standards* (PERMIS) website. Available at: http://sec.cs.kent.ac.uk/permis.

[14] Shibboleth website. Available at: http://shibboleth.net.

[15] Audun Jøsang , John Fabre , Brian Hay , James Dalziel , Simon Pope. Trust Requirements in Identity Management. Proceedings of the 2005 Australasian workshop on Grid computing and e-research, Darlinghurst, Australia, Australian Computer Society, Inc. (2005) 99–108.

[16] Bhargav-Spantzel, A.; Squicciarini, A.C.; Bertino, E., "Trust Negotiation in Identity Management," *Security & Privacy, IEEE* , vol.5, no.2, pp.55,63, March-April 2007 doi: 10.1109/MSP.2007.46.

[17] J Camenisch, S Moedersheim, D Sommer. A Formal Model of Identity Mixer. Formal Methods for Industrial Critical Systems, 198--214, Springer, 2010.

[18] J Camenisch, R Leenes, M Hansen, J Schallaboeck. An introduction to privacy-enhancing identity management. Digital privacy, 3--21, Springer, 2011.

[19] Khattak, Z.A.; Sulaiman, S.; Manan, J.A., "A study on threat model for federated identities in federated identity management system," Information Technology (ITSim), 2010 International Symposium in , vol.2, no., pp.618,623, 15-17 June 2010, doi: 10.1109/ITSIM.2010.5561611.

[20] Gail-Joon Ahn; Sekar, P., "Ontology-Based Risk Evaluation in User-Centric Identity Management," *Communications (ICC), 2011 IEEE International Conference on* , vol., no., pp.1,5, 5-9 June 2011, doi: 10.1109/icc.2011.5962948.

[21] Lei Jin; Takabi, H.; Joshi, J.B.D., "Security and Privacy Risks of Using E-mail Address as an Identity," *Social Computing (SocialCom), 2010 IEEE Second International Conference on* , vol., no., pp.906,913, 20-22 Aug. 2010, doi: 10.1109/SocialCom.2010.134.

[22] Ammar Alkassar and Rani Husseiki. D3.9: Study on the Impact of Trusted Computing on Identity and Identity Management. Fidis Project Deliverable D3.9, 2008.

[23] White, P., "Identity Management Architecture: A new direction," *Computer and Information Technology, 2008. CIT 2008. 8th IEEE International Conference on* , vol., no., pp.408,413, 8-11 July 2008, doi: 10.1109/CIT.2008.4594710.

[24] Rieger, S., "User-Centric Identity Management in Heterogeneous Federations," *Internet and Web Applications and Services, 2009. ICIW '09. Fourth International Conference on* , vol., no., pp.527,532, 24-28 May 2009, doi: 10.1109/ICIW.2009.85.

[25] Authentication & Authorization Infrastructure of the Max Planck Society. Available at: https://aai.mpg.de.

[26] Secure idenTity acrOss boRders linKed (STORK) 2.0. Available at: https://www.eid-stork2.eu

[27] Neisse, Ricardo, Alexander Pretschner and Valentina Di Giacomo. "A Trustworthy Usage Control Enforcement Framework," International Journal of Mobile Computing and Multimedia Communications (IJMCMC) 5 (2013): 3, doi:10.4018/jmcmc.2013070103.

[28] Neisse, R.; Doerr, J., "Model-based specification and refinement of usage control policies," Privacy, Security and Trust (PST), 2013 Eleventh Annual International Conference on , vol., no., pp.169,176, 10-12 July 2013, doi: 10.1109/PST.2013.6596051.

[29] D. Quartel, "Action relations - basic design concepts for behaviour modelling and refinement," PhD Thesis University of Twente, 1998.

[30] OpenID Authentication 2.0, Available at http://openid.net/specs/openid-authentication-2_0.html

[31] OAuth, http://oauth.net/documentation/getting-started/

[32] C. Neuman, T.Yu, S. Hartman, K.Raeburn, The Kerberos Network Authentication Service (V5), RFC 4120

[33] E. Rescorla, Diffie-Hellman Key Agreement Method, RFC 2631

[34] Secure Apache HBase, http://hbase.apache.org/book/security.html

[35] Using external authentication with Keystone, http://docs.openstack.org/developer/keystone/external-auth.html

[36] Omar Sefraoui, Mohammed Aissaoui and Mohsine Eleuldj. Article: OpenStack: Toward an Open-source Solution for Cloud Computing. International Journal of Computer Applications 55(3):38-42, October 2012.

[37] Birrell, E.; Schneider, F.B., "Federated Identity Management Systems: A Privacy-Based Characterization," Security & Privacy, IEEE , vol.11, no.5, pp.36,48, Sept.-Oct. 2013

[38] Blaze, M., Ioannidis, J., Keromytis, A.D.: Experience with the keynote trust man-agement system: Applications and future directions. In: iTrust. (2003) 284{300}

[39] Winslett, M.: An introduction to trust negotiation. In: iTrust. (2003) 275{283}

[40] Seamons, K.E., Winslett, M., Yu, T., Smith, B., Child, E., Jacobson, J., Mills, H., Yu, L.: Requirements for policy languages for trust negotiation. In: POLICY.(2002) 68{79}

[41] Bertino, E., Ferrari, E., Squicciarini, A.C.: Trust-x: A peer-to-peer framework for trust establishment. IEEE Trans. Knowl. Data Eng. 16(7) (2004) 827{842}

[42] Braghin, S., Fovino, I., Trombetta, A.: Advanced trust negotiation in critical infrastructures. In: Infrastructure Systems and Services: Building Networks for a Brighter Future (INFRA), 2008 First International Conference on. (2008) 1{6}

[43] ISO: Information Technology Security Techniques A Framework for Identity Management-Part 1 (2011)

[44] Installation Guidelines for OpenStack, http://docs.openstack.org/folsom/basic-install/content/basic-install_requirements.html

European Commission

EUR 26523 EN – Joint Research Centre – Institute for the Protection and Security of the Citizen

Title: Electronic Soft-Identities (E-Ids) State-of-the-art and Multi-Morphed E-Ids, an Explorative Study

Author(s): Igor NAI FOVINO, Ricardo NEISSE, Dimitris GENEIATAKIS, Ioannis KOUNELIS

Luxembourg: Publications Office of the European Union

2014 – 52 pp. – 21.0 x 29.7 cm

JRC87768

EUR 26523 EN

ISSN 1831-9424

ISBN 978-92-79-35529-5

DOI 10.2788/78078

Abstract :

The scientific contribution of this report is twofold: on a side it provide a first, explorative overview of the state of the art in the world of soft-identity management systems, and on the other he present the first outcomes of a research activity aiming at proposing a solution to address trust and privacy protection issues related to identity and personal data provided by citizens in a smart environment. Our proposed solution combines identity management, trust negotiation, and usage control. The concept of identity management allows creation of less privacy sensitive soft identities derived from hard identities with high assurance. Trust negotiation techniques are used during the authentication phase to support the identity establishment process between the entities in the smart city. After the identity is established we use usage control policies to govern the exchange of identity and personal data in a privacy friendly manner.

LB-NA-26523-EN-N

As the Commission's in-house science service, the Joint Research Centre's mission is to provide EU policies with independent, evidence-based scientific and technical support throughout the whole policy cycle.

Working in close cooperation with policy Directorates-General, the JRC addresses key societal challenges while stimulating innovation through developing new standards, methods and tools, and sharing and transferring its know-how to the Member States and international community.

Key policy areas include: environment and climate change; energy and transport; agriculture and food security; health and consumer protection; information society and digital agenda; safety and security including nuclear; all supported through a cross-cutting and multi-disciplinary approach.

Publications Office