González Manzano, L., González-Tablas, A. I., De Fuentes, J.M., Ribagorda, A. (2014). Extended U+F Social Network Protocol: interoperability, reusability, data protection and indirect relationships in Web Based Social Networks. *Journal of systems and software*, 94, pp. 50-71

DOI: https://doi.org/10.1016/j.jss.2014.04.044

Extended U+F Social Network Protocol: Interoperability, reusability, data protection and indirect relationships in Web Based Social Networks

Lorena González-Manzano*, Ana I. González-Tablas, José M. de Fuentes, Arturo Ribagorda

Computer Science and Engineering Department, University Carlos III of Madrid, Avda. de la Universidad, 30, 28911 Leganés, Spain

A B S T R A C T

An interconnected world is what current technologies look for, being Web Based Social Networks (WBSNs) a promising development in this regard. Four desirable WBSN features are identified, namely, interoperability, reusability, protection against WBSNs providers and indirect relationships. A protocol, called U+F, addressed interoperability and reusability of identity data, resources and access control poli-cies between different WBSNs. In order to address the remaining couple of features, that is, achieving the protection of data against WBSNs providers and indirect relationships management across different WBSNs, this paper presents eU+F, an extension of U+F. A prototype is developed to verify the feasibility of implementing the proposed protocol in a real environment, as well as to compare its workload regarding three well-known WBSNs, Facebook, MySpace and LinkedIn.

Keywords:
Data disclosures
Interoperability
Web Based Social Networks

1. Introduction

From recent years until present time much more than a hundred of Web Based Social Networks (WBSNs) have emerged, being Facebook, MySpace, Badoo and LinkedIn some representative examples. Users are eager to interact with their contacts, even considering them friends, families or just work partners. They desire to share their experiences either by interchanging elements such as photos or videos or using specific applications to satisfy their expectations.

Given the quantity and assorted purposes of WBSNs, users want to interact with people no matter the WBSN in which they are enrolled, thereby attaining *interoperability and reusability* according to resources, identity data and access control policies. Resources mainly correspond to photos, videos and audio files and identity data refers to profile and contact relationship data. Specifically, interoperability refers to achieve data management between different WBSNs and reusability can be identified as a complementary feature to interoperability because if a pair of elements are interoperable between multiple WBSNs, it means that they can be analogously used and, thus, they can be reused. Multiple proposals look for interoperability, like OpenID[1] which bases on identity data or LotusNet that focuses on resources interoperability (Aiello and Ruffo, 2012).

Moreover, most of WBSN users look for new people to whom establish some kind of relationship, without necessarily being direct contacts. Indeed, indirect relationships are an inherent property of the society, as C. Calhoun noticed (Acquisti and Gross, 2006), society is a question of social integration where the growing relevance of *indirect relationships* is related to modernity. Thus, indirect relationships in WBSNs correspond to the number of jumps that users can perform from one user to others, also called depth (Carminati et al., 2006, 2007), and their establishment is essential.

On the other hand, the protection of data against unnoticed or non-consented uses is other desirable feature. There have been several attempts to conceal data from servers (Jammalamadaka et al., 2008; di Vimercati et al., 2007) and, regarding recent trends, it is referred as *data exposure minimization* (Ciriani et al., 2011). In the great majority of cases, when registering in a WBSN it is mandatory to accept the established privacy policy. Multiple WBSN privacy policies specify the management and use of all uploaded data. An extremely related example is the new Google's privacy policy in

* Corresponding author. Tel.: +34 916249113; fax: +34 916249129.
E-mail address: lgmanzan@inf.uc3m.es (L. González-Manzano).

[1] http://openid.net/ (last accessed October 2013).

which the use of all users' data to improve experience in Google applications is detailed (Google Team, 2012).

A previous protocol achieved interoperability and reusability between WBSNs combining the application of User-Managed Access (UMA) protocol and the Friend-Of-A-Friend (FOAF) project (González-Manzano et al., 2012). UMA refers to an architecture and protocol to give web users control over who and what can get access to their online personal data (Kantara Members, 2009). By contrast, the latter, FOAF, provides guidelines to develop files describing personal data and relationship among different users (FOAF Team, 2000). This previous protocol was called UMA + FOAF Social Network Protocol (U+F).

In order to address the remaining couple of features, particularly, data exposure minimization and indirect relationships management across different WBSNs, this paper presents Extended UMA + FOAF Social Network Protocol (eU+F). eU+F is an extension of U+F which combines UMA and FOAF, together with cryptographic techniques. Given that eU+F extends U+F, their underlying architecture is similar. It must be noted that entities involve in the architecture are well-known in the security domain and they usually appear in access control architectures (Anderson, 2001). Thus, the main contribution of this paper is a protocol which protects the stored resources while providing interoperability among different WBSNs and enabling a proper management of user-to-user relationships for access control purposes.

The evaluation of eU+F is performed at two different levels, theoretically to verify the satisfaction of the proposed requirements and experimentally to analyse the protocol workload using a prototype. Besides, the experimental evaluation includes a comparison between eU+F workload and that of three of the most successful WBSNs, namely, Facebook, MySpace and LinkedIn. Thanks to the conducted evaluation (which is much more comprehensive than that of U+F) it is possible to assess the suitability of the proposal in practical settings.

The paper is structured as follows. Section 2 describes related works. Section 3 introduces the set of works which lay the bases of this paper. Section 4 presents a general overview of eU+F. Section 5 describes the system model, involving the requirements to attain, the trust and adversary models and the proposed architecture. Section 6 presents a detailed description of the protocol. In Section 7 a pair of cryptographic approaches to manage data exposure minimization are defined. Section 8 presents an analysis regarding the satisfaction of requirements. In Section 9 a theoretical evaluation of the protocol is presented. Section 10 describes the experimental evaluation, including the developed prototype and results achieved. Section 11 presents a discussion concerning improvements that can be performed on eU+F to reach a powerful approach. Finally, in Section 12 conclusions and open research issues are presented.

2. Related work

Interesting and multiple proposals focus on providing certain kind of interoperability between WBSNs. In general, they present a particular social network structure within which data and access control policies are managed following a specific pattern, including some of them the use of cryptographic techniques. Table 1 presents the results of the analysis, identifying per each studied proposal if it addresses interoperability/reusability, data exposure minimization and/or indirect relationship management.

A great amount of proposals address interoperability and they particularly manage direct relationships. The most related approach is Lockr (Tootoonchian et al., 2009), which runs in centralized WBSNs and decentralized peer-to-pear (P2P) systems. It bases on the interchanged of tokens, referred as social attestations, that certify the relationship between a issuer and a receiver. LotusNet (Aiello and Ruffo, 2012) is other challenging approach, it consists of a P2P system in which peers store resources locally and it relies on cryptography to guarantee strong authentication and confidentiality through the use of distributed hash tables. Other noticeable work is PrPl (Seong et al., 2010), a decentralized architecture based on locating personal data on chosen hosts, called *Personal butlers* and the management of data through the interchange of tickets between contacts. Similarly, Conti et al. (2011) introduce Virtual Private Social Networks (VPSN). These particular social networks consist of storing personal data in a personal server and sending fake information to WBSNs like Facebook. Then, special xml files are sent to users contacts to retrieve the real information. Furthermore, a recent approach is proposed by Riesner and Pernul (2012). They specify a global model which points out the need of managing data independently of WBSNs, known as Social Identity Management (SIdM), attaining that users control in great depth their identity data.

By contrast, a relevant amount of works applies cryptographic techniques and deal with data exposure minimization. Two different sets of proposals are identified, those focused on direct relationships management and those oriented to the indirect ones. Regarding direct relationships, which are addressed by the majority of authors, FlyByNight (Lucas and Borisov, 2008) is a challenging

Table 1
Related work description.

Proposals	Requirements		
	Interoperability/reusability	Data exposure minimization	Indirect relationships
Lockr (Tootoonchian et al., 2009)	√		
LotusNet (Aiello and Ruffo, 2012)	√		
PrPl (Seong et al., 2010)	√		
VPSN (Conti et al., 2011)	√		
SIdM (Riesner and Pernul, 2012)	√		
FlyByNight (Lucas and Borisov, 2008)		√	
NOYB (Guha et al., 2008)		√	
LifeSocial.KOM (Graffi et al., 2010)		√	
Persona (Baden et al., 2009)		√	
Prometheus (Kourtellis et al., 2010)		√	√
Key allocation (Frikken and Srinivas, 2009)		√	√
OpenID[a]	√		
Diaspora[b]		√	
Scramble[c]		√	

[a] http://openid.net/ (last accessed October 2013).
[b] http://diasporaproject.org/ (last accessed October 2013).
[c] https://addons.mozilla.org/en-US/firefox/tag/scramble (last accessed October 2013).

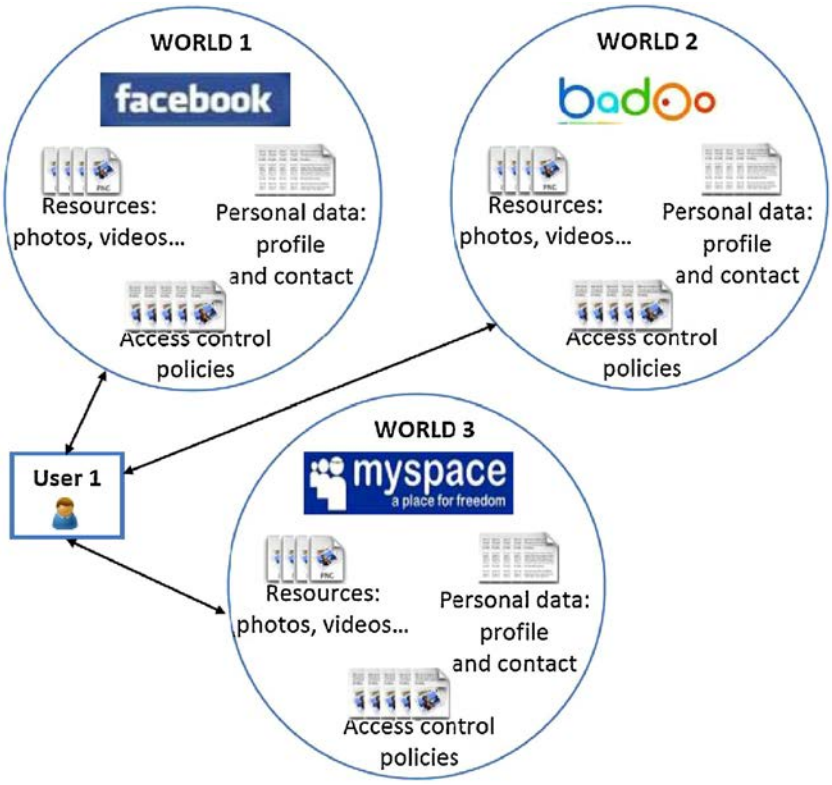


Fig. 1. Current WBSNs.

development. It focuses on posted messages which are presented encrypted in the WBSN, Facebook in this case, and they only are decrypted using the appropriate decryption keys. Applying different techniques, Guha et al. (2008) present NOYB, an approach to encrypt data applying a pseudo-random substitution cipher from a public dictionary. Other remarkable proposal is LifeSocial.KOM (Graffi et al., 2010), a P2P architecture in which resources are stored encrypted through a symmetric algorithm and the decryption process is perform through an asymmetric algorithm. A different cryptographic algorithm is applied in Persona (Baden et al., 2009) that manages access control through the application of attribute based encryption cryptography. In particular, the main issue refers to the use of attributes in the establishment and creation of keys and access control policies.

On the other hand, according to indirect relationships a pair of approaches are noticed. Prometheus (Kourtellis et al., 2010) bases on a P2P service which recollects encrypted data from multiple devices. Users are connected through a social graph in which nodes corresponds to trusted peers who store data encrypted and edges refer to tags that represent access control policies. From a different perspective, Frikken and Srinivas (2009) bases on exclusively managing access control by the establishment of access control policies based on jumps between users. Moreover, it mainly focuses on avoiding data exposure minimization making use of a particular cryptographic technique which consists of using, per each user, as many keys as the maximum number of jumps accepted. Furthermore, different to all previous contributions, in this proposal not only can key management be performed by WBSN users, but also by the server which stores data.

Apart from academic approaches, some interesting proposals are currently developed in the professional world. In relation to the search of interoperability, OpenID[2] is noticed. It is an open protocol that allows authentication across different platforms. Although it has been supported by some WBSNs, i.e. MySpace, it mainly bases on interoperability in terms of identity data. Diaspora[3] is other crucial example. It is a distributed social network based on a P2P architecture in which peers stored their personal resources in a particular host and remain them available for their contacts. Besides, applying cryptographic algorithms to encrypt stored data, data exposure minimization is also addressed. However, though the real focus of Diaspora is to achieve interoperability, currently, this WBSN exclusively interacts with Diaspora servers, called *pods*. Also looking for protecting data against WBSNs providers, Scramble[4] is a remarkable solution. It is a Firefox plugin which enforce users access control preferences through the application of cryptographic techniques to all data uploaded to the web.

Considering the previous analysis, neither of the previous proposals manage indirect relationships (of an unlimited depth) while looking for interoperability as well as preventing data from being disclosed by WBSNs without users consent. This need motivates the protocol proposed in this paper.

3. Background

A WBSN is described as a large quantity of users connected between each other that manage and access multiple and assorted data. On the one hand, users who upload data and manage them are referred as administrators and users who request access to data are called requesters. On the other hand, data correspond to resources, such as photos, and videos, a personal profile, contacts relationships and access control policies linked to resources and the personal profile. These data are stored in each WBSN where users are enrolled.

[2] http://openid.net/ (last accessed October 2013).

[3] http://diasporaproject.org/ (last accessed October 2013).

[4] https://addons.mozilla.org/en-US/firefox/tag/scramble (last accessed October 2013).

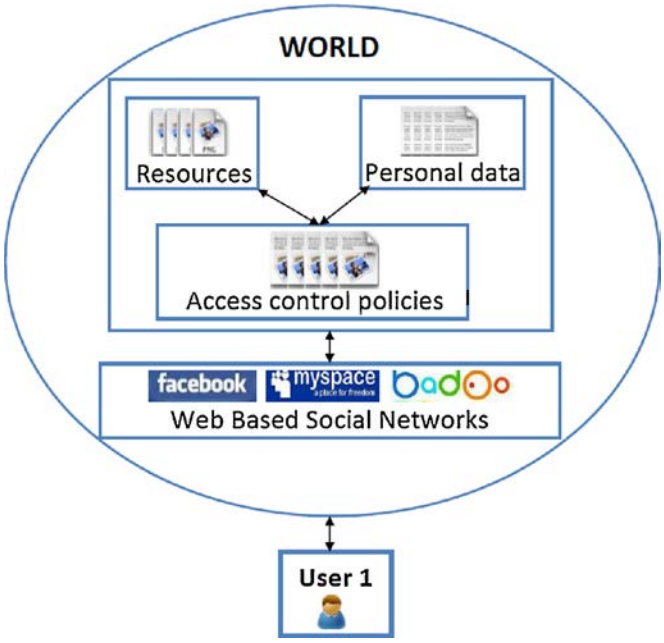


Fig. 2. U+F Social Network Protocol.

Due to that fact, WBSNs can be identified as different worlds which centralize the management and storage of data, as Fig. 1 depicts.

Nonetheless, the centralization of data makes difficult managing interoperability and reusability and looking for these requirements satisfaction U+F was developed (González-Manzano et al., 2012). The focus of U+F is the creation of a single world where all kind of WBSNs are interoperable between each other. In this protocol resources, identity data (profile and contacts relationships) and access control policies are located out of WBSNs to be reused and to simplify access control management (Fig. 2). In this section UMA and FOAF are described together with a brief summary of U+F to establish the basis for the work presented herein.

3.1. User-Managed Access (UMA) specification

The UMA architecture and core protocol (Machulak et al., 2010; Hardjono, 2012), based on OAuth, provides a dedicated access relationship service in different web domains where users are able to modify the conditions of access and terminate relationships easily. UMA provides key features to achieve resources and access control policies interoperability and reusability between different services because resources are stored in Hosts and access control policies in AMs, thus facilitating the access from different services to resources and access control policies.

Several UMA implementations[5] have been developed but they are not related to WBSNs. There are one commercial UMA authorization server and a total of three publicly available projects, particularly, the Fraunhofer AISEC project (which offers a client, an AM and a Host currently running), the OXAuth project (that facilitates the implementation of UMA for enterprise usages) and the SMART project (which involves the implementation of UMA together with sample applications).

3.2. Friend-Of-A-Friend (FOAF) specification

FOAF is a project which provides a machine-readable ontology to describe people, things they create and do, and links between them (FOAF Team, 2000). It combines the use of the Resource Description Framework (RDF) and the Web Ontology Language (OWL). More specifically, the FOAF specification provides guidelines to structure and develop files in which personal data, such as name, phone, homepage, interests or photos or known users, like friends or relatives, are described.

Identified in Carminati et al. (2009), FOAF seems a promising approach in regard to the specification of user identity within the WBSNs' context. Multiple WBSNs, such as Twitter, and social applications, like Second Life, make use of it.[6,7]

3.3. U+F Social Network Protocol

UMA + FOAF Social Network Protocol (U+F) is a novel development to manage interoperability and reusability between WBSNs (González-Manzano et al., 2012). Interoperability management is named in the literature as the *Wall Garden Problem* (Yeung et al., 2009). It refers to the inability of WBSNs to work together within and across any type of boundary in order to advance the effective communication of all users. It is associated with the access from different WBSNs to resources, identity data and access control policies. Moreover, reusability is another issue managed in U+F. As aforementioned, if elements are interoperable between WBSNs, they can be equally used and, also, reused.

The essential purpose of U+F is the acquisition of identity data and resources either personal or of a direct contact enrolled in the same or in a different WBSN. UMA is applied to decentralize resources and access control policies and FOAF to decentralized identity data.

3.3.1. Personal file

Identity data is composed of users' profiles and contacts' relationships data. They are structured and stored in FOAF files. As described in González-Manzano et al. (2012), attributes "nationality", "WBSNs", "creation date", "trust" and "duration" are at stake.

Note that, in U+F, reduced FOAF files are also used, called in (Ackermann et al., 2009) sub-profiles. These files contain less data than original FOAF files and they are applied in the access control enforcement process.

3.3.2. Architecture

U+F is composed of six types of entities which are described as follows.

1. *User* (U): A user plays different roles. On the one hand, a user plays the role of a UMA's Requesting Party (RP) who is able to access resources of his contacts through WBSNs. On the other hand, a user also plays the role of an Authorizing User (AU) by locating resources in his Host, his FOAF file in his Identity Provider (IdP) and established policies in his Authorization Managers.
2. *Identity provider* (IdP): Repository of FOAF files which are placed by AUs, as well as provider of claims. This entity can be compared with a Host but instead of storing resources, it stores identity data. Besides, to manage claims, per each user, IdPs store a list of IdP Certification Authorities (IdP_CAs) that each user considers reliable. Moreover, to guarantee communications with WBSNs that are trusted by users, per user, a list of WBSN Certification Authorities (WBSN_CAs) which are considered trustworthy is also stored.
3. *Host*: Repository of resources, analogous to a data base service, in which the AU stores resources.

[5] https://kantarainitiative.org/confluence/display/uma/UMA+Implementations?src=contextnavchildmode.

[6] http://www.xul.fr/web-2.0.html (last accessed February 2014).

[7] http://www.w3.org/wiki/FoafSites (last accessed February 2014).

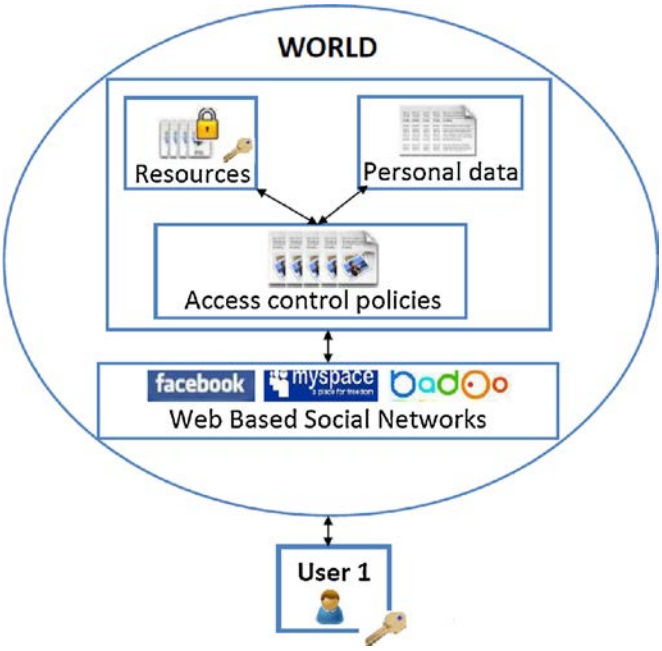


Fig. 3. eU+F Social Network Protocol.

4. *Authorization manager* (AM): Entity that evaluates policies previously established by an AU. However, to achieve this purpose the AM requests claims to perform policy validation and delivers tokens. Also, in order to verify claims, they store, per each user, a list of the IdP_CAs trusted by the user. Likewise, to communicate with WBSNs considered trusted by users, per user, a list of WBSN_CAs which are considered reliable is also stored.
5. *Web Based Social Networks*: Provide an interface to show resources and identity data and also, provide the management of wall comments, resource comments and any other extra services. Moreover, this entity acts on behalf of a RP and interacts with Hosts to reach protected resources; interacts with AMs to get the appropriate token in regard to requested resources; and interacts with the adequate IdP to get users' personal data each time a user session starts. Each WBSN owns a certificate generated by a WBSN Certification Authority (WBSN_CA).
6. *Certification authorities* (CA): These entities are in charge of delivering certificates to trusted entities to allow them signing interchanged messages. A pair of groups are distinguished. A first group provides certificates to IdPs (IdP_CAs) and another group to WBSNs (WBSN_CAs). Then, per user, AMs and IdPs store a list of IdP_CAs to ensure, along the protocol execution, that claims are provided from trusted IdPs. Likewise, IdPs and AMs store, per user, a list of WBSN_CAs to ensure that interoperability is only allowed between trusted WBSNs.

4. System overview

Recalling Section 2, a couple of demanding necessities are recognized, being both of them out of the scope of U+F (Section 3.3). First, as in current WBSNs, *indirect relationships* have to be managed. Second, data is out of users control and WBSNs can use it for their own purposes without users consent, being the protection of this issue called *data exposure minimization*. In order to face up these new challenges a more powerful and secure protocol is proposed in this work, Extended UMA + FOAF Social Network Protocol (eU+F) (see Fig. 3). From a more specific point of view, in eU+F identity data corresponds to the profile and contacts of each user and it is stored in the form of FOAF files within Identity Providers (IdPs)

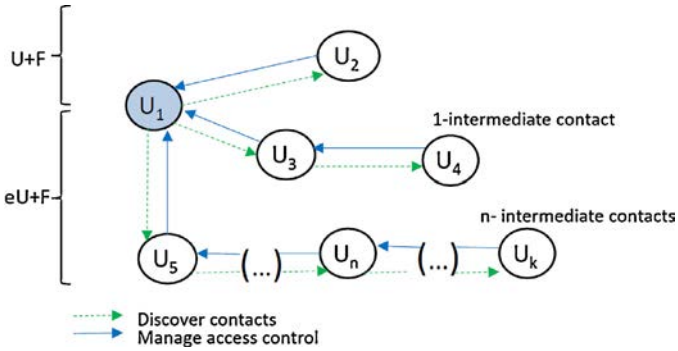


Fig. 4. Managed relationships.

(as in U+F); resources are stored encrypted in chosen Hosts (as in U+F but being encrypted in this extension); and access control policies are located in Authentication Managers (AMs) which perform access control on behalf of the users (as in U+F).

Regarding access control management, this protocol applies an access control system based on an Extension of $UCON_{ABC}$ access control model, called $SoNeUCON_{ABC}$ (González-Manzano et al., 2013, 2014). This model mainly basis on managing subjects, objects and relationships attributes. Moreover, it is specially focused on relationships management either unidirectional, bidirectional (composed of a pair of directional ones), direct, indirect or any other type. Although within $SoNeUCON_{ABC}$ model access control can be based on any type of relationship between the requester and the administrator of the requested data, within eU+F protocol, for simplicity reasons and analogous to U+F, only bidirectional relationships between the administrator of the requested data and the requester are considered. Then, for instance, if $User_A$ wants to access to a resource of $User_B$, it is required that $User_A$ has specified having a relationships with $User_B$, as well as $User_B$ has specified having a relationship with $User_A$. These direct relationships must be stored in each user's FOAF file.

Furthermore, in eU+F, regarding the discovery of indirect contacts, although it could be done from scratch (i.e. selecting an unknown user and searching between an indirect relationship between him and the requester), it is assumed that users access data of indirect contacts that can be reached from their direct contacts.

Despite aforementioned, the mechanism proposed herein can be used for any type of relationship between the administrator and the requester of a particular data and also, other indirect contacts discovery procedure can be applied. These generalizations are discussed in Section 11. Following, the way in that eU+F proceeds is illustrated with an example (see Fig. 4). Analogously to U+F, assuming a direct relationship between U_2 and U_1, and the fact that U_1 wants to access U_2's data, the access is granted if the relationship is bidirectional and a proof of the existence of the relationship U_2–U_1 is obtained from IdP_U_2, that is, U_1 is within U_2's contacts (solid arrow). On the other hand, given the management of indirect relationships proposed in eU+F, supposing that U_1 has already accessed to U_2's profile (including his direct contacts) and U_1 wants to access U_4's data, the access is granted if there exist bidirectional relationships between all involved users in the path and it is obtained from IdP_U_4 a proof of the existence of a relationship between U_4–U_1. This proof is constructed step by step. First, IdP_U_3 certifies the relationship U_3–U_1 (solid arrow) and then, after presenting this proof to IdP_U_4, this IdP certifies the relationship U_4–U_3 (solid arrow). Finally, the proof U_4–U_1 is constructed. Therefore, it is clearly noticed that access control bases on the existence of relationships in the opposite direction to the discovery of contacts. However, it is remarkable that getting the proof is not enough to get access because it depends on access control policies and thus,

not only the proof has to be obtained but also policies have to be satisfied. Finally, note that the set of relationships that are managed in eU+F are the most practical ones for efficiency reasons because it is known, in each step, the next contact of the relationship.

Regarding eU+F working plan, it bases on the acquisition of identity data and resources of WBSN users. The overall idea behind the acquisition of these data are analogous to U+F but needing, first, the management of proofs (elements that compose claims) to verify the existence of indirect relationships and, second, the use of cryptography to deal with data exposure minimization. Then, the acquisition of identity data and resources is summarized as follows. Once a user logins in a WBSN, data are requested to the appropriate IdP or Host and it redirects to the necessary AM to verify policies attached to the requested data. After policies have been properly verified (to perform this verification appropriate claims should be presented), the AM grants or denies access delivering or not a ticket, called token, to be presented to the data storage. Then, if the token is valid, data are granted. Lastly, as granted data are encrypted, its decryption is performed at users browsers. More specifically, eU+F is composed of four phases. The first phase is the **initialization**. It refers to the configuration of entities and elements involved in the protocol. Subsequently, the second phase starts when a **user logins in a WBSN**. At this moment, the user accesses to his identity data and contacts data which are stored in the chosen IdP. Besides, his resources, stored in a particular Host, remain available. The third phase is the **access to data of a direct contact** who is enrolled in a different WBSN. In particular, it is divided in accessing to the contact's identity data and resources. Finally, **access to an indirect contact** enrolled in another WBSN (different from any other) is the last phase. It is also divided in the acquisition of identity data and resources and it works similar to access to a direct contact data but requiring a proof to verify the existence of the appropriate indirect relationship.

5. System model

The model involves the specification of requirements (Section 5.1), the trust and adversary model (Section 5.2) and the architecture (Section 5.3).

5.1. Requirements

Regarding eU+F features, the following requirements are challenges to attain:

1. **Data confidentiality and access control**. Data has to be exclusively delivered and used by authorized users and entities involved in the protocol. Moreover, it is noticeable that the access control has to be performed by the management of relationships between administrators and requesters.
2. **Interoperability and reusability regarding direct and indirect relationships**.The communication and interchange of data between multiple users enrolled in different WBSNs has to be attained.
3. **Chain of trust**.Given the great set of entities at stake, the final receiver has to be able to verify that entities through which interchanged messages pass are trusted.
4. **Data privacy preservation against WBSNs**.Data has to be adequately protected from WBSNs and to achieve it, this requirement is subdivided as follows:
   (a) **Data exposure minimization**.A particular set of data has to remain inaccessible to WBSNs, being protected against inappropriate managements. Furthermore, it is desirable that Hosts do not get access to data.
   (b) **Accessibility to minimum data**. The amount of data accessible to WBSNs has to be minimized. Once a WBSN accesses to data of a user of any WBSN, the management has to be carried out using the least possible data. Indeed, this is directly related with "The principle of least privilege" which bases on the fact that every programme should operate using the least possible amount of privileges (Saltzer and Kumar, 1975). In particular, this is called data minimization (Borcea-Pfitzmann et al., 2011) and it can be identified as a common principle in the development of Privacy Enhancing Technologies.
5. **Simple key management**. Keys have to be easily managed, which means that decryption keys are not distributed out of band such as it is done in Baden et al. (2009) or in Guha et al. (2008) because, due to the large amount of users, the distribution can become unmanageable.

5.2. Trust and adversary model

In the social networking world the adversary is directly related to applications that illegitimately manage personal information. In this protocol, resources, identity data and access control policies are decentralized and WBSNs do not store and directly manage them. However, WBSNs act as interfaces which present requested data.

Regarding these features, the trust model bases on the following considerations:

- IdPs and AMs are trusted entities. In other words, these entities do not maliciously manipulate data and they perform operations following strictly eU+F specifications.
- Hosts are untrusted entities. They may use data for their own purposes but carry out operations regarding rigorous eU+F specifications.
- WBSNs are considered untrusted entities. They may use data for their own purposes, as well as, they may try to act on behalf of the user when he is not logged in the WBSN. Nonetheless, given that the goal of this protocol is to attain interoperability between WBSNs, all of these applications have to fairly manage all messages involved in the protocol. On the contrary, if a particular WBSN alters messages content or produces incorrect message deliveries, it would damage its own business model, leading the rest of WBSNs and users who trust it to lose their confidence.
- Analogous to many web applications in which personal data are managed, communications between entities are carried out through a confidential and mutual-authenticated channel, such as SSL.

Regarding previous assumptions the adversary corresponds to a WBSN that acquires and illegitimately manipulates and uses identity data and resources of its users for multiple purposes (leak information to external parties). For instance, WBSNs can use data for advertising or, even worst, for trading with other companies. Additionally, these applications are in charge of sending and receiving messages to and from IdPs, Hosts and AMs on behalf of users and consequently, WBSNs can obtain as much information as possible (though encrypted) when a user is not really logged. Similarly, Hosts store resources and they may use them for their own purposes.

5.3. Architecture

The architecture of eU+F is composed by the same groups of entities as those of U+F (recall Section 3.3.2). The main differences are the addition of a new set of certification authorities for AMs

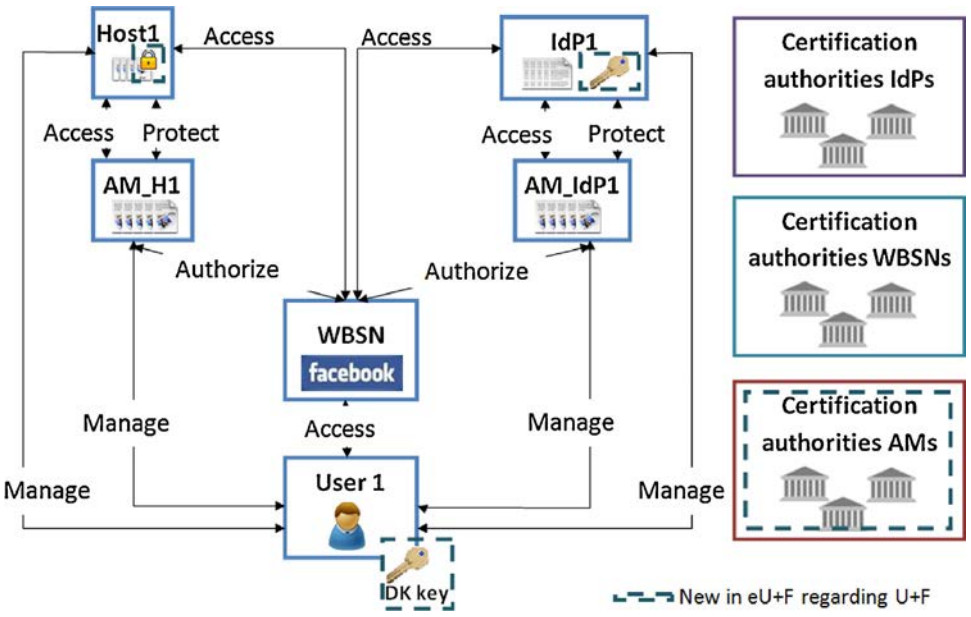


Fig. 5. Architecture.

(see Fig. 5), and the introduction of new tasks for existing entities. Thus, introduced changes are explained below:

1 *User*: In eU+F, user resources are stored symmetrically encrypted using a key called resources decryption key (DK). Besides, they have to also establish the necessary relationships between their Hosts and their IdPs and their AMs. In addition, each user is in charge of creating, at least, a symmetric key used in the encryption and decryption of resources and an asymmetric key pair (which can correspond to a private key and the associated public key certificate or to a created key pair) to manage decryptions and interchanges of encrypted data. These keys are used in the schemes described in Section 7 to attain data exposure minimization.
2 *Identity provider (IdP)*: IdPs are now responsible for creating reduced FOAF files regarding stored FOAF files and access control policies processed by AMs. In other words, IdPs store complete FOAF files but certain attributes and relationship data can be accessible to some users and denied to others, being necessary the creation of FOAF files, called reduced FOAF files, which only include data in respect to access control policies satisfaction.
3 *Host*: Concerning hosts, stored resources have to be periodically re-encrypted, either by the host under the users' supervision or directly by users who update the data re-encrypting it with a new key. Notice that re-encryptions require the update of the used key in the appropriate IdP.
4 *Authorization manager (AM)*: This entity owns a certificate and the associated private key to sign claims. Then, trustworthiness of requested claims is guaranteed.
5 *Web Based Social Network (WBSN)*: In eU+F, these entities provide a viewer, used by their users, to perform the decryption and presentation of resources and identity data. Besides, they may provide other extra services such as wall comments, resource comments or adds management which are offered as in a common web applications like current WBSNs. Indeed, these services are the focus of the WBSNs' business model, marking differences among multiple WBSNs.
6 *Certification authorities (CA)*: In the proposed protocol, a new group of certification authorities is introduced (AM_CA) which provides certificates to AMs that comply to a set of CA-defined rules. Furthermore, Hosts and IdPs can refer to personal servers of WBSNs users or servers of particular companies.

The motivation of having three sets of CAs is twofold. First, it is convenient in a real-world scenario since it is the most simple setting from the administrative point of view. Second, it is beneficial from the security point of view since it responds to the separation of duties principle. Nevertheless, cross-certification and mutual recognition agreements could exist among them, as it happens in real-life deployments.

6. eU+F protocol description

The description of eU+F is divided in the definition of messages content (Section 6.1), the definition of the execution procedure, that is, the phases involved in the protocol (Section 6.2), and the specification of concrete differences between the execution procedure in eU+F and U+F (Section 6.3).

6.1. Messages content

Along the protocol an assorted set of messages is interchanged. More specifically, messages content corresponds to operations, elements and structures.

6.1.1. Operations

Operations involved in eU+F correspond to signatures and encryptions. Moreover, in respect to encryption, symmetric and asymmetric algorithms are applied according to the cryptographic schemes proposed in Section 7 to deal with data exposure minimization.

6.1.2. Elements

In general, there are six elements within interchanged messages: *user identifiers*, which refer to emails stored in a hashed way; *tokens*, that have attached an expiration time; *file identifiers*, that identify identity data (FOAF files) and resources; *tickets*, that identify a requested data and are used to get access tokens; *signatures*, that include a time stamp and are specially significant in claims management attesting the existence of users attributes and

Table 2
Interchanged messages in eU+F.

Id	Name	Content
M1	Token request	Ticket $\|\|WBSN_{R\_}Cert\_Serial\_Number\|\|\ Date\_time_{WBSN_{Rsignature}}\ \|\|$ $S_{k_{WBSN_{R\_}Cert}}$ (complete message)
M2	Token request redirection	Ticket $\|\|AM\_location$
M3	Token response redirection	Ticket $\|\|Tokenvalue\|\|Expired-in\|\|AM_{A\_}Cert\_Serial\_Number\|\|Date\_time_{AM_{Asignature}}\|\|$ $S_{k_{AM_{A\_}Cert}}$ (complete message)
M4	Token response	Token response redirection $\|\|WBSN_{R\_}Cert\_Serial\_Number\|\|\ Date\_time_{WBSN_{Rsignature}}\|\|$ $S_{k_{WBSN_{R\_}Cert}}$ (complete message)
M5	File request	R_Id $\|\|A\_Id\|\|File\_Id\|\|WBSN_{R\_}Cert\_Serial\_Number\|\|\ Date\_time_{WBSN_{Rsignature}}\|\|$ $S_{k_{WBSN_{R\_}Cert}}$ (complete message)
M6	File indirect request	R_Id $\|\|A\_Id\|\|File\_Id\|\|WBSN_{R\_}Cert\_Serial\_Number\|\|Date\_time_{WBSN_{Rsignature}}\|\|$ $S_{k_{WBSN_{R\_}Cert}}$ (complete message)
M7	File response	R_Id $\|\|A\_Id\|\|E_{k_R}$(file)
M8	Claims request	R_Id $\|\|A\_Id\|\|E_{k_{CertIdP_R}}\ (Data_R request)\|\|AM_A\_Cert\_Serial\_Number\|\|$ $Date\_time_{AM_A signature}\|\|S\_k\_CertAM\_A$ (complete message)
M9	Claims structures response	R_Id $\|\|A\_Id\|\|AccreditationR\|\|E_{k_{CertAM_A}}(Data\_Rresponse)\|\|IdP_R\_Cert\_Serial\_Number\|\|$ $Date\_time_{IdP_R signature}\|\|S_{k_{IdP_R}}$ (complete message)
M10	Claims response	Claims structures response $\|\|RelationshipR\_A_1\|\|IdP_A\_Cert\_Serial\_Number\|\|$ $Date\_time_{IdP_A signature}\|\|S_{k_{IdP_A}}(RelationshipR\_A\_1)\|\|WBSN_{R\_}Cert\_Serial\_Number\|\|$ $Date\_time_{WBSN_{Rsignature}}\|\|S_{k_{WBSN_{R\_}Cert}}$ (complete message)
M11	Certify direct relationship	R_Id $\|\|A\_Id\|\|AccreditationR\|\|IdP_R\_Cert\_Serial\_Number\|\|Date\_time_{IdP_R signature}\|\|$ $S_{k_{IdP_R}}(AccreditationR)\|\|RelationshipR\_A_1\|\|WBSN_{R\_}Cert\_Serial\_Number\|\|$ $Date\_time_{WBSN_{Rsignature}}\|\|S_{k_{WBSN_{R\_}Cert}}$ (complete message)
M12	Certify indirect relationship	R_Id $\|\|A\_Id\|\|AccreditationR\|\|IdP_R\_Cert\_Serial\_Number\|\|Date\_time_{IdP_R signature}\|\|$ $S_{k_{IdP_R}}$ (Accreditation R) $\|\|RelationshipR\_A_i\|\|IdP_{A\_i}\_Cert\_Serial\_Number\|\|$ $Date\_time_{IdP_{A_i} signature}\|\|S_{k_{IdP_{A_i}}}\ (RelationshipR\_A\_i)\|\|RelationshipR\_A_1\|\|$ $WBSN_A\_Cert\_Serial\_Number\|\|Date\_time_{WBSN_A signature}\|\|S_{k_{WBSN_{A\_}Cert}}$ (complete message)
M13	Relationship certified	R_Id $\|\|A\_Id\|\|RelationshipR\_A_1\|\|IdP_A\_Cert\_Serial\_Number\|\|Date\_time_{IdP_A signature}\|\|$ $S_{k_{IdP_A}}$ (Relationship R_A_1)
M14	Simple token request	Ticket
M15	Simple token response	Ticket $\|\|Tokenvalue\|\|Expired-in\|\|AM_A\_Cert\_Serial\_Number\|\|Date\_time_{AM_{Asignature}}\|\|$ $S_{k_{AM_{A\_}Cert}}$ (complete message)
M16	Simple file request	R_Id $\|\|$ File_Id
M17	Simple file response	R_Id $\|\|E_{k_R}$(file)
M18	Simple claim request	R_Id $\|\|A\_Id$
M19	Simple claim response	R_Id $\|\|IdP_{R\_}Cert\_Serial\_Number\|\|AccreditationR\|\|Date\_time_{IdP_R signature}\|\|$ $S_{k_{IdP_R}}$ (Accreditation R)
M20	Token validation	Ticket $\|\|Tokenvalue$

relationships; and *redirections*, which refer to URLs that point out the location of the entities to which redirections are performed.

6.1.3. Structures

Structures correspond to sets of elements over which operations are performed. In eU+F there are four main types of applied structures: the *Accreditation* which identifies who is the requester of a particular requested file; the *RelationshipA-B*$_i$, that refers to the identifiers of the users involved in a relationship, where *i* refers to the number of jumps that separate both users; and the *Data request* and the *Data response* that are used to verify the satisfaction of each established access control policy. The former is provided by the requester and consists of the name of the attributes involved in the applied policy, that is, *attributes: att1 att2 att3 . . .*, and the latter refers to the values of all requested attributes within a *Data request*, that is, *attributes: att1 att2 att3 . . . attributesData: valueAtt1 valueAtt2 valueAtt3*.

Messages interchanged in the eU+F protocol are depicted in Table 2 where symbol || implies concatenation, *S* refers to signature and *E* to encryption. This table presents each message content in regard to operations, elements and structures aforementioned. Nonetheless, technical details of interchanged messages are pointed out in Appendix A. Interchanged messages mainly follow UMA's core protocol specification (Hardjono, 2012), although some new fields have been added in some cases and a few new messages have been specified.

6.2. Execution procedure

Recalling the protocol phases described in Section 4, eU+F is divided in four phases: the **initialization** phase, in which the initialization of entities is performed; **User logins in a WBSN**, in which a user, in the role of a RP, logins in a WBSN and accesses to his encrypted identity data and resources, being data locally decrypted; **User accesses to data of a direct contact** where a user, also in the role of a RP, tries to access to the profile and resources of a direct contact who is registered in a different WBSN, being data locally decrypted; and **User access to data of an indirect contact** in which a user, again in the role of a RP, accesses to data of an indirect user who is registered in a different WBSN (data are also locally decrypted). It is remarkable that accessing a direct or an indirect contact data within the same WBSN follows the same procedure as accessing data of a user enrolled in a different one. Most protocol phases in eU+F already existed in U+F. However, in order to address the new requirements, multiple changes are needed within each phase. For the sake of clarity, each phase will be explained including U+F and eU+F related issues. Introduced modifications are summarized in Section 6.3.

6.2.1. Initialization

In this phase entities are prepared with all required data. It consists of the *Registration of resources and identity data*, the *Registration of entities* and the *Specification of main information in WBSNs*.

Registration of resources and identity data. In this phase resources are located in chosen Hosts and the appropriate FOAF file in the chosen IdP. This pair of tasks is analogous to that of UMA (Machulak et al., 2010). Moreover, the user, in the role of an AU, symmetrically encrypts resources using a DK key and uploads them to chosen Hosts. Similarly, identity data together with the used DK are uploaded to chosen IdPs.

Registration of entities. This phase consists of the establishment of a trust relationship between a Host or an IdP and an AM. In particular, it involves the registration of a Host in an AM and the registration of an IdP in an AM, which can be the same AM or a different one. These registrations are equivalent to the introduction of Host to AM described in UMA (Machulak et al., 2010).

The following step is the creation and establishment of access control policies in chosen AMs. Furthermore, the registration process continues specifying in AMs the list of trusted IdP_CAs and WBSN_CAs, in IdPs the list of trusted IdP_CAs, AM_CAs and WBSN_CAs and in Hosts the list of trusted WBSN_CAs. To conclude, the user defines in his IdPs, Hosts and AMs the list of WBSNs with which he desires to interact. He also establishes in his IdPs, AMs and WBSNs an accepted time stamp threshold in order to control signatures expiration time.

Specification of main information in WBSNs. Once a user enrols in a WBSN, he has to specify the IdP in which his FOAF file is stored and the Host which stores his resources. Likewise, to allow interactions between WBSNs, in each of them the set of WBSNs with which communications are available has to be established. Looking at users' expectations it would be desirable to maximize the amount of WBSNs that can interact between each other.

6.2.2. User logins in a WBSN

When a user, in the role of a RP, logins in a WBSN three processes are carried out, his authentication, the acquisition of his profile and contacts and the acquisition of his resources which remain accessible but are not directly presented. The step of accessing a protected resource of the UMA protocol (Machulak et al., 2010; González-Tablas et al., 2010) is executed a couple of times: the WBSN, in the role of a requester and on behalf of the user, contacts first the user's IdP to get his FOAF file and second, the user's Host to get his resources.

Firstly, users can delegate in WBSNs to access to his data. However, before the delegation, the user authenticates himself against his Host and IdP to inform that he is logged in the WBSN which can act on his behalf until his log-out. Guaranteeing that WBSNs do not act on behalf of users when they are logged out requires performing the authentication procedure in the log in and out in the WBSN, thereby informing the user's IdP and Host that he is or is not connected.

After finishing the authentication, following the UMA protocol, the profile and contacts of the user (the FOAF file) are directly presented and his resources remain accessible to be retrieved when desired. Claims and tokens used to get the FOAF file can be reused to get resources. Indeed, claims, that consist of the user's email hash properly signed (called herein *accreditation*), are stored in WBSNs along each user session to be repetitively used until they expire. Nonetheless, it should be noticed that data is locally decrypted.

6.2.3. User accesses to data of a direct contact

A user, once logged in a WBSN, may want to access to data of other user, that is, to a contact's data. In case both users are not enrolled in the same WBSN, the communication between their WBSNs becomes essential. Assuming that a user of WBSN1, User1, desires to access to resources of User2, enrolled in WBSN2, all WBSNs in which User2 is registered in are identified. User2's identifier, his email, and the set of WBSNs in which he may be registered in are available in the FOAF file of User1, as described in Section 3.3.1. Then, User1 chooses a WBSN and the procedure described in this section is executed.

Afterwards, User1 clicks on the relationship with User2 and if there exists a relationship between User2 and User1, User2's profile and resources are delivered in regard to access control policies. Note that relationships are unidirectional and the correctness of the relationship direction has to be verified before granting access to data. It is not the same that User2 has a relationship with User1 (User1 is included in User2's FOAF file) than the other way round.

Similar to the login phase, the first step is the acquisition of the FOAF file from the appropriate IdP and the second step the acquisition of resources from the appropriate Hosts. Again, the step of accessing a protected resource of UMA protocol is executed a couple of times, one to acquire the FOAF file of User2, and another to obtain resources of User2 (which may be performed repetitively). Although, the fact that data are locally decrypted has to be recalled. Besides, regarding the previous example, it should be noticed that applied claims are composed of three structures: (P1) a proof of the valid existence of User1; (P2) a proof of the relationship between User2 and User1, verifying that User1 relationship is within the FOAF file of User2; and (P3) a proof of User1 being in possession of the set of data required to validate access control policies. Therefore, to get P1 and P3, WBSN1 contacts to IdP_User1. By contrast, to obtain P2, IdP_User1 creates P2*, which is marked with * because the existence of a relationship between User2 and User1 has not been certified by the appropriate IdP. Then, P2* is sent to IdP_User2 which proves the validity of P2* and provides P2 if the verification is successful.

6.2.4. User accesses to data of an indirect contact

Considering the existence of indirect relationships, the procedure is rather similar to the one described in Section 6.2.3 except for requiring interactions between all WBSNs involved in the relationship. In particular, WBSN interactions are indispensable to acquire claims that prove the existence of an indirect relationship between a pair of users. For instance, given three users such that User1 is directly connected to User2 and User2 to User3, to verify the indirect relationship between User3 and User1 it is necessary to request a proof of the existence of such relationship to IdP_User3. Then, the request sent to IdP_User3 attaches a proof of the relationship between User2 and User1 and IdP_User3 verifies if User3 has a relationship with User2 to finally certify the indirect relationship between User3 and User1. Nonetheless, it is noteworthy that apart from getting the proof, User3's access control policies have to be satisfied to get the requested access.

As in *User accesses to data of a direct contact* (Section 6.2.3), the procedures of acquiring identity data and resources are quite analogous. Indeed, the main difference is that IdPs provide identity data and Hosts provide resources. Consequently, recalling the previous example and considering that this phase is the most challenging one, the following section describes the acquisition of User3's FOAF file.

FOAF file acquisition. The acquisition of a FOAF file requires as many UMA executions as WBSNs are involved in the relationship minus one. In general, this procedure focuses on recursively repeating the acquisition of FOAF files from each of the WBSNs involved the indirect relationship. Indeed, this feature is what points out that an indirect relationship can be defined as multiple direct ones. Thus, according to the proposed example two execution of UMA are performed, the first one to get the FOAF file of User2 and the second one to get the FOAF file of User3. To better understand this phase the procedure is depicted in Fig. 6 and, in brackets, message identifiers in regard to Table 2 are noticed.

More specifically, after having acquired identity data of User2, applying the procedure described in Section 6.2.3, User1 clicks on User2 relationship with User3. Afterwards, User1 chooses to access

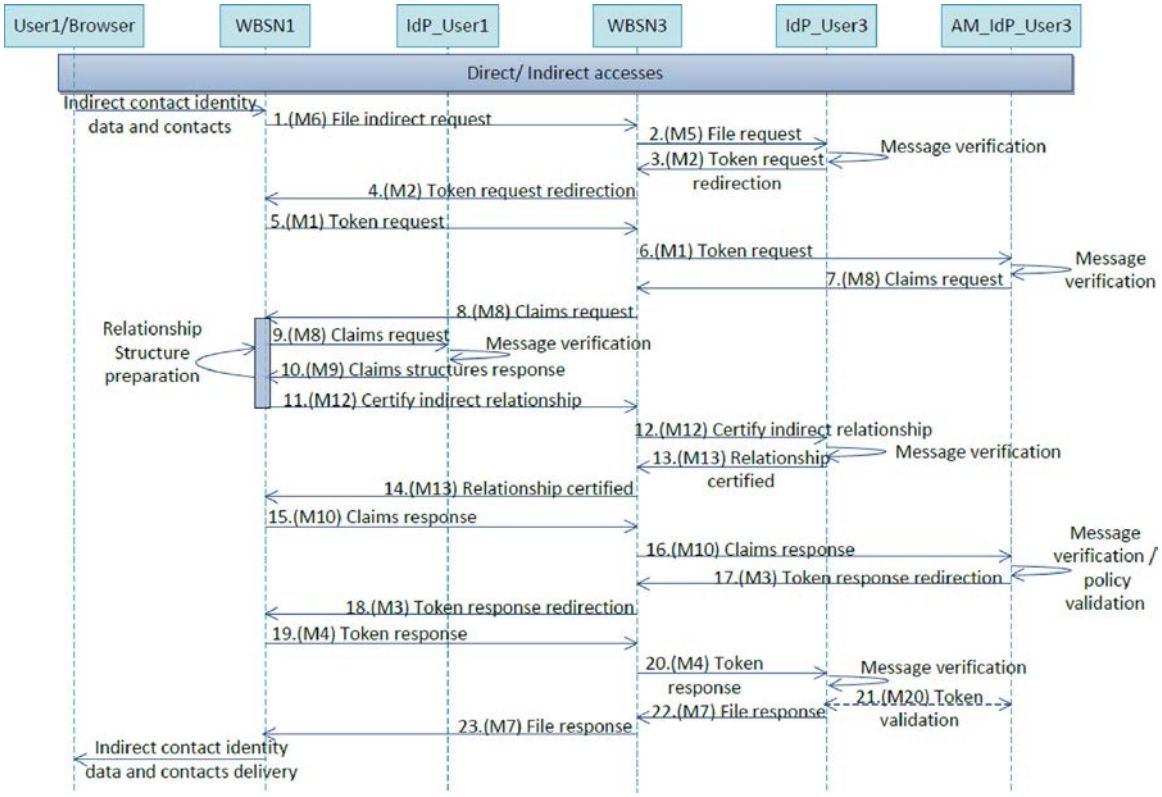


Fig. 6. User accesses to the FOAF file of an indirect contact.

to User3's identity data and an the protocol described next is carried out. WBSN1 interacts with WBSN3 and it requests to IdP_User3 the User3's FOAF file (msg. 1 and 2 of Fig. 6). Subsequently, IdP_User3 requests an access token and redirects WBSN1 to AM_IdP_User3 (msg. 3–6). Then, AM_IdP_User3 requests claims (msg. 7 and 8) that are analogous to the ones requested when accessing a direct contact except for P2 which has to proof the existence of the indirect relationship between User3 and User1. Therefore, P1 is reused and P3 is reused or requested depending on requested claims (msg. 9 and 10). By contrast, obtaining P2 requires the interaction with WBSN3. Indeed, WBSN1 creates P2* that corresponds to a non-certified proof of the relationship between User3 and User2 and sends it together with the P2 previously obtained (while accessing to User2's data) that certifies the relationship between User2 and User1 to IdP_User3 (msg. 11 and 12). The IdP_User3 verifies the existence of the relationship, creates the new P2 and sends it back (msg. 13,14). When WBSN1 gets claims (composed of P1, P2 and P3), sends them to AM_IdP_User3 and if their verification is successful the access token is delivered (msg. 15–18). Lastly, the token is sent to IdP_User3 and the requested file is provided (msg. 19–23). However, the IdP delivers an encrypted reduced FOAF file and it has to be decrypted in the user's browser applying one of the schemes proposed in Section 7.

6.3. *Identifying differences: eU+F vs U+F*

UMA and the FOAF project lay the bases of U+F and eU+F. In consequence, both protocols share a significant set of elements. Table 3 compares the number of involved entities, signatures, signatures verifications and messages interchanged in each protocol, being differences of eU+F put in bold. Identified from the table, in eU+F more signatures are required, a new group of entities is added (AM Certification Authorities, AM_CAs), C_A, and a new execution phase for indirect relationships management is introduced.

In order to clearly notice the distinction between tasks and phases of eU+F in contrast to U+F, differences regarding the protocol execution procedure are detailed as follows:

- **Initialization** (Section 6.2.1). On the one hand, users have to create a set of keys. Moreover, the specification of lists of trusted WBSN_CAs in AMs, trusted AM_CAs and WBSN_CAs in IdPs and trusted WBSN_CAs in Hosts is required. On the other hand, users have to store in chosen Hosts their resources, encrypted, and the symmetric keys applied in the resources encryption in their IdPs.
- **User logins in a WBSN** (Section 6.2.2). In eU+F the decryption of users identity data and resources is performed locally, at users browsers, following one of the schemes described in Section 7.
- **User accesses to data of a direct contact** (Section 6.2.3). Again, one of the cryptographic schemes proposed in Section 7 has to be applied.
- **User accesses to data of an indirect contact** (Section 6.2.4). This phase is a new one since indirect relationships were out of the scope of U+F.

Furthermore, messages content of U+F and eU+F differs to a great extent. Comparing interchanged messages in eU+F with those interchanged in U+F, the following features are distinguished:

- Complexity of claims management increases because it is required the accreditation of an indirect relationship.
- Tokens and claims requests are signed by the appropriate AMs to guarantee the trustworthiness of requested data.
- Requested claims are encrypted by AMs and decrypted by IdPs. Conversely, IdPs encrypt requested claims and AMs decrypt them. Then, this issue protects identity data from being accessible to WBSNs.
- Messages signed by AMs include the AM certificate serial number to identify the signer entity.

Table 3
Theoretical comparison U+F vs eU+F.

Phases		Entities	Signatures	Signatures verification	# Messages
Login					
FOAF file acquisition	U+F	$3 + C_I + C_{WBSN}$	1	1	12
	eU+F	$3 + C_I + C_{WBSN} + \mathbf{C_A}$	**2**	**2**	12
Resource acquisition	U+F	$3 + C_I + C_{WBSN}$	1	1	12
	eU+F	$3 + C_I + C_{WBSN} + \mathbf{C_A}$	**2**	**2**	12
Access to direct contact					
FOAF file acquisition	U+F	$6 + C_I + C_{WBSN}$	6	6	25
	eU+F	$6 + C_I + C_{WBSN} + \mathbf{C_A}$	**8**	**8**	25
Resource acquisition	U+F	$7 + C_I + C_{WBSN}$	6	6	25
	eU+F	$7 + C_I + C_{WBSN} + \mathbf{C_A}$	**8**	**8**	25
Access to indirect contact					
FOAF file acquisition	U+F	–	–	–	–
	eU+F	$3 \cdot (N+1) + C_I + C_{WBSN} + C_A$	$8 \cdot (N+1)$	$8 \cdot (N+1)$	$25 \cdot (N+1)$
Resource acquisition	U+F	–	–	–	–
	eU+F	$3 + 4 \cdot N + C_I + C_{WBSN} + C_A$	$8 \cdot (N+1)$	$8 \cdot (N+1)$	$25 \cdot (N+1)$

N, (# of users in the relationship) − 1, $N > 1$; C_x, # of IdP_CAs, AM_CAs and WBSN_CAs, where x is I, WBSN or A regarding the type of CA; –, an element/action not required.

- Depending on the applied cryptographic approach (see Section 7), the interchange of the decryption key in the Traditional PKC scheme or the decryption key creation in the IBE-based PKC scheme, is required to get access to resources.
- Resources and identity data are delivered encrypted to be decrypted at users' browsers, thereby preventing WBSNs from accessing users' data.

7. Data exposure minimization management

There are multiple possibilities, making use of cryptography, to prevent WBSNs from visualizing resources and identity data presented in them. However, regarding one of the security requirements, decryption keys cannot be distributed off-line because, as WBSNs are used by a huge quantity of users and lots of them are not directly known, the procedure would be impractical. Therefore, a hybrid encryption approach, similar to (Graffi et al., 2010), is applied to resources management and an asymmetric one to identity data management. In particular, a pair of alternatives to manage and distribute keys are described in the following sections, one of them focuses on traditional Public Key Cryptography (PKC) and the other one focuses on PKC based on Identity Based Encryption (IBE). However, it is an open issue the election of a particular algorithm.

Furthermore, it is essential to consider advantages and disadvantages of achieving data exposure minimization. The main advantage is to prevent WBSNs from using personal data for their own purposes such as sending spam or building profiles of users likes and dislikes. Nonetheless, there are some drawbacks to highlight. Firstly, the time required to perform the protocol increases due to the cryptographic operations applied. Second, users are in charge of encrypting their resources and uploading them and the applied key. Third, a particular amount of extra storage is required to store keys. Finally, several messages are added to the protocol, such as those for providing the decryption keys. Next, Sections 7.1 and 7.2 describe the application of PKC and IBE and Section 7.3 presents a comparison of the application of both schemes in eU+F.

7.1. Traditional PKC

This technique bases on the well-known concept of PKC (Salomaa, 1996). Each user owns a key pair (K_{pub} and K_{pv}), or multiple ones.

In the *Initialization* phase each user delivers his K_{pub} with his FOAF file and his resources decryption key, DK, to the preferred IdP. Then, acquisition of identity data focuses on requesting the appropriate K_{pub} and use it to encrypt and retrieve the requested FOAF file. On the other hand, resources, encrypted with DK, are retrieved and decrypted using K_{pub} to reach DK. The use of this mechanism involves introducing some new messages apart from those already present in Section 6.1. To get a better picture of interchanged messages, Fig. 7 depicts the acquisition of resources, where (E) points out messages that already exist, (I) refers to messages that are new but can be included within existing ones and (N) points out new messages that have been created from scratch.

7.2. IBE-based PKC

To reduce the burden of key management Identity Based Encryption (IBE) cryptography is applied (Boneh and Franklin, 2001). The primary innovation of IBE is the use of user identity attributes, e.g. email, address and so on, for encryption and signatures verification. K_{pub} are created from public parameters together with chosen user identity attributes. By contrast, K_{pv} are generated by trusted third parties, called herein IBE authorities. Depending on the algorithm, the creation of the public key pair may require the interaction with IBE authorities to establish some common variables. Therefore, it is recommendable the use of an algorithm like (Maurer and Yacobi, 1996) which focuses on exclusively creating the public key through public parameters without depending on an additional number chosen by a user or by an authority. Likewise, the complexity and quantity of involved IBE authorities depends on the applied proposal (Joe, 2009). The most significant advantage of IBE is the unnecessary use of a public key infrastructure, thereby avoiding certificates management and reducing the system complexity and the cost for establishing and managing keys. Due to that fact revocation is also simplified. When attributes change, i.e. emails, K_{pv} does not have to be re-distributed or replaced in a public repository.

Assuming that eU+F uses the users' email as an identity attribute, once the attached K_{pv} is provided by an IBE authority, the acquisition of identity data and resources is analogous to the PKC technique except for not requiring the delivery of K_{pub} in IdPs.

7.3. Comparison: traditional PKC vs IBE-based PKC

This section analyses advantages and disadvantages of traditional PKC and IBE-based PKC schemes. A summary of the analysis is presented in Table 4.

Both schemes present a pair of significant common advantages in regard to key management. Firstly, on-line key interchanges are not required. This is essential in applications like WBSNs because users share data among multiple contacts and key distribution may

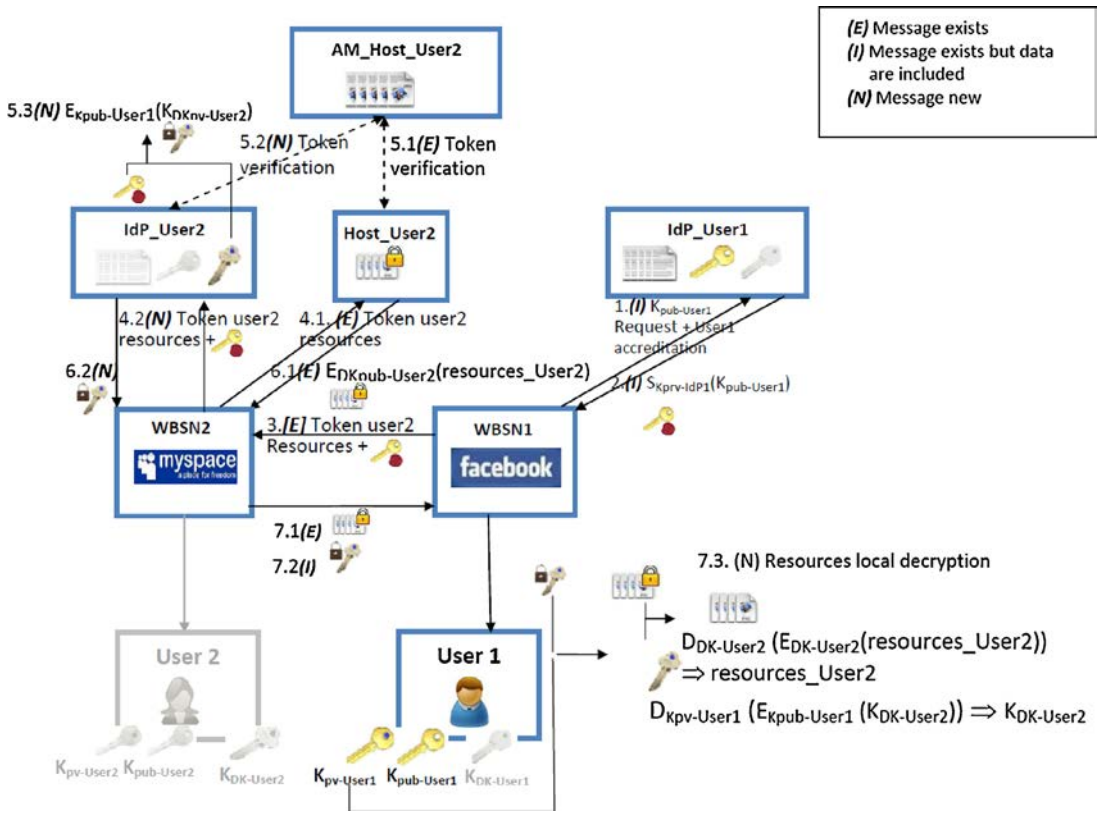


Fig. 7. Traditional PKC – Acquiring User2's resources.

Table 4
PKC vs IBE-based PKC.

	Traditional PKC	IBE-based PKC
Simple key management	√	√
Extra entities		√
Extra storage	√	
Impact performance		√

be a burden. Second, keys can be periodically updated preventing attacks, in the traditional PKC scheme, against the applied encryption algorithm and in the IBE-based PKC scheme, against the applied IBE algorithm. Indeed, in this last scheme, the update of keys may involve the change of parameters in the used IBE key creation algorithm.

Concerning the traditional PKC scheme, it has the advantage of not involving extra entities in the protocol. Besides, this scheme presents the huge benefit of not affecting the protocols performance to a great extent, that is, resources are symmetrically encrypted and just decryption keys management uses asymmetric cryptography. By contrast, in the IBE-based PKC scheme, apart from involving high computational operations, IBE authorities have to take part in the protocol. Nonetheless, these new entities release the necessity of extra storage space for public keys, as well as the necessity of being IdPs in change of their delivery.

8. Requirements evaluation

eU+F has to fulfil all proposed requirements described in Section 5.1. Therefore, each of them is analyzed, identifying their level of satisfaction.

Concerning **data confidentiality and access control**, this protocol bases on UMA and adds a concrete claims management. In a nutshell, access control bases on the satisfaction of access control policies after proving the appropriate claims, that include a proof of the existence of a relationship between the administrator of the requested data and the requester. Afterwards, a token, with a particular expiration time, is delivered according to the satisfied policies and then, access is granted until the token expires. Moreover, when the token is presented to IdPs or Hosts to get the requested data, if the token verification defined in UMA involves verifying that the entity which presents the token is the same one to which the token was initially delivered, nobody except for authorized users get access. By contrast, if the UMA implementation does not verifies this issue, an adversary (internal or external) may reach the token and use it until it exceeds. However, this matter directly depends on UMA and it is out of the scope of this paper.

In addition, WBSNs may act on behalf of users and get access to data while the user is logged (note that data are encrypted). Therefore, to prevent WBSNs from acting on behalf of users when they are not logged, each user authenticates himself against his Host and IdP in the log-in and log-out in the WBSN.

The second requirement is **interoperability and reusability in respect to direct and indirect relationships**. It is achieved due to the decentralization of identity data, resources and access control policies management, being all of them stored in IdPs, Hosts and AMs respectively. Data can be replaced, moved or updated without affecting any service of WBSNs. Moreover, different WBSNs can make use of the same resources, identity data and access control policies if the same IdPs, Host and AMs are linked to them. More specifically, regarding interoperability, the use of the same identity data specification, FOAF files in this case, and the use of a concrete application of UMA, including the specification of claims and the Fat Requester, addresses this issue.

Other requirement is **chain of trust**. Users establish in their IdPs, Hosts, AMs and WBSNs the appropriate lists of trusted IdP_CAs, AM_CAs, WBSN_CAs and WBSNs. Then, the most relevant interchanged messages, that are sent between multiple entities, are

Table 5
eU+F theoretical evaluation: protocol phases.

Phases	Entities	Encryptions	Decryptions	Signatures	Signatures verification	# Messages
Initialization	$I+H+A+1$	R	*	*	*	$12\cdot(I+H)+2\cdot A$
Entities registration	$I+H+A+1$	*	*	*	*	$10\cdot(I+H)+2\cdot A$
Registration of resources and identity data	$I+H$	R	*	*	*	$2\cdot(I+H)$
Specification of information in WBSNs	$1+S$	*	*	*	*	*
Login	$6+(C_1\cap C_2)$	–	1	4	4	24
Authentication	*	*	*	*	*	*
FOAF file acquisition	$3+C_1$	–	–	2	2	12
Resource acquisition	$3+C_2$	–	1	2	2	12
Access to direct contact	$8+(C_1\cap C_2)$	4	5	16	16	50
FOAF file acquisition	$6+C_1$	2	2	8	8	25
Resource acquisition	$7+C_2$	2	2	8	8	25
Access to indirect contact	$3+5\cdot N+(C_1\cap C_2)$	$4\cdot(N+1)$	$4\cdot(N+1)$	$16\cdot(N+1)$	$16\cdot(N+1)$	$50\cdot(N+1)$
FOAF file acquisition	$3\cdot(N+1)+C_1$	$2\cdot(N+1)$	$2\cdot(N+1)$	$8\cdot(N+1)$	$8\cdot(N+1)$	$25\cdot(N+1)$
Resource acquisition	$3+4\cdot N+C_2$	$2\cdot(N+1)$	$2\cdot(N+1)$	$8\cdot(N+1)$	$8\cdot(N+1)$	$25\cdot(N+1)$

N, (# of users in the relationship) − 1, $N>1$; I, # of IdPs of a user; A, # of AMs of a user; *, an element/action not detailed; R, # of resources of a user; H, # of Hosts of a user; C_x, set of IdP_CAs, AM_CAs and WBSN_CAs; –, an element/action not required.

signed by issuer entities as well as by entities though which they pass to finally verify that signer entities are within the stored lists, that is, they are trusted. In particular, messages related to the acquisition of claims are properly signed by IdPs and AMs and messages interchanged between WBSNs are signed by the WBSN at which the user wants to access to.

The following key requirement is **data privacy preservation against WBSNs** which is divided in two. On the one hand, WBSNs cannot access to users' data because data as well as decryption keys are encrypted and their decryption is performed at users' browsers, achieving **data exposure minimization**. Indeed, resources are stored encrypted and identity data and decryption keys are encrypted when they are delivered, being all of them locally decrypted. Moreover, it has to be noticed that the local decryption must be performed under security constraints, that is, decrypted data cannot leave the user's browser. Furthermore, it is remarkable the fact that Hosts store encrypted resources and, analogous to WBSNs, they have not got access to data.

On the other hand, **access to the minimum data** is related to claims management. For example, the easiest way would be the interchange of complete FOAF files between WBSNs. Nevertheless, to satisfy the proposed requirement, data interchanged between WBSNs is limited to users identifiers, WBSNs in which they are enrolled and, in case of indirect relationships, proofs that certify the relationship between the administrator of the requested data and the requester. Therefore, this requirement is satisfied to a very large extent, though leaving as an open issue that WBSNs know the relationships proofs.

To conclude, the last requirement to analyze is **simple key management**. The proposed cryptographic schemes suppose the creation of as many asymmetric key pairs and symmetric keys as it is desired. However, assuming that keys are indispensable in any cryptographic approach, in the proposed schemes they are not interchanged out of band and it simplifies their management. Indeed, removing out of band interchanges of keys prevents from possible management confusions either intentionally or not. Besides, considering the large quantity of WBSNs and the amount of established relationships, out of band interchanges may become unmanageable.

9. Theoretical evaluation

This evaluation analyses, in Section 9.1, the workload of each protocol phase presented in Section 6.2, and, in Section 9.2, that of the schemes proposed to deal with data exposure minimization presented in Section 7. Specifically, it is evaluated the number of entities involved, the number of encryptions and decryptions carried out, the number of signatures and signatures verification performed and the number of messages interchanged. Results are summarized in Tables 2 and 3.

9.1. Protocol phases analysis

Table 5 analyses each protocol phase. Note that the reuse of claims and tokens is not considered, as well as data exposure minimization techniques which are studied in the following section (Section 9.2).

Regarding the amount of entities the protocol involves, the use of IdPs, Hosts and AMs is particularly noticeable in the *initialization* because relationships between all entities that interact along the protocol are established in this phase. Besides, a significant amount of entities come into play when *accessing data of an indirect contact*, that is, the longer the relationship, the higher the number of involved entities.

Encryption is applied for a couple of issues. On the one hand, in the *initialization* resources are encrypted and uploaded to chosen Hosts. On the other hand, encryption protects the delivery of claims. AMs encrypt data involved in requested claims and IdPs encrypt such requested data to be sent to AMs. Besides, it should be noticed that the number of encryptions increases when *accessing data of an indirect contact* because more claims are requested.

Related to encryption, decryptions are executed at claims management and at resources acquisition. In Section 9.2, related to data exposure minimization, cryptographic operations are deeply analyzed.

Signatures are other elements at stake. They are applied to verify the chain of trust which is created between entities that interchanged messages. Signatures are performed by AMs when requesting claims, by IdPs when delivering claims and by WBSNs when sending messages to other WBSNs. Again, the number of signatures increases when *accessing data of an indirect contact* because more claims and interactions among WBSNs are carried out.

Following expectations, the number of signatures verification is equivalent to the number of signatures. In general, IdPs, AMs and WBSNs make signatures and IdPs and AMs verify them.

Last but not least, the amount of messages involved in eU+F is remarkable. It is specially significant the number of interchanged messages *accessing data of an indirect contact*. Nonetheless, it can decrease reusing tokens and claims because the reuse avoids requesting tokens to AMs and claims to IdPs. Likewise, the reuse also decreases the number of signatures and encryptions.

Table 6
eU+F theoretical evaluation: data exposure minimization.

Phases	Entities	Encryptions	Decryptions	Signatures	Signatures verification	# Messages
Traditional PKC	–	2	3	2	2	3
FOAF file acquisition	–	1	1	1	1	–
Resources acquisition	–	1	2	1	1	3
IBE-based PKC	$C_1 \cap C_2$	2	3	–	–	3
FOAF file acquisition	C_1	1	1	–	–	–
Resources acquisition	C_2	1	2	–	–	3

C_x, set of IBE authorities; –, an element/action not required.

9.2. Data exposure minimization analysis

An analysis of the cryptographic alternatives described in Section 7 is performed distinguishing the acquisition of identity data and the acquisition of resources. Results are presented in Table 6. Note that this study bases on cryptographic matters and it is not attached to the rest of eU+F messages.

Regarding entities involved in *Traditional PKC* and *IBE-based PKC*, it is the latter technique which applies a new group of entities, called IBE authorities.

According to encryption, both techniques require the same number of operations. IdPs create FOAF files and encrypt them once delivered. Likewise, resources acquisition involves the encryption of resources decryption keys.

On the other hand, the number of decryptions acquiring FOAF files and resources differs. FOAF files acquisition simply bases on decrypting requested files. By contrast, resources decryption requires, first, decrypting the resources decryption key and subsequently, applying this key to decrypt the resources.

In relation to signatures, they are only applied in *Traditional PKC* when acquiring requesters' public keys. These keys are signed and delivered by requesters' IdPs to be properly verified by administrators' IdPs.

Finally, both techniques involve the interchange of three new messages. These messages are used to get resources decryption keys.

10. Experimental evaluation

The experimental evaluation corresponds to the analysis, from a practical point of view, of eU+F through the development of a prototype. First, Section 10.1 presents the architecture, purpose and technical details of the developed prototype. Second, Section 10.2 presents the experimental results regarding the measurement of the protocol temporal workload and its comparison with Facebook, MySpace and LinkedIn.

10.1. eU+F prototype

This section presents the development of a prototype to prove the viability of implementing eU+F in a simulated environment. It is composed of two WBSNs, FriendBook+and MyLeisure. The general architecture is depicted in Fig. 8. More specifically, a couple of IdPs, a couple of Hosts, a couple of WBSNs and four AMs (one for each Host and IdP) are the entities at stake. Thus, a total of eight servers are used and located in different places along a local network. The key point is to verify that data of MyLeisure remains available to FriendBook+and the other way round. Users' identity data (profile and contacts) corresponds to their name, nationality, age, email, school and contacts relationships. For simplicity reasons, the prototype only works with direct relationships. However, obtained performance results allow us to get estimated figures of performance of the protocol for indirect relationships, because as noticed in Section 6.2.4, an indirect relationship can be defined as multiple direct ones.

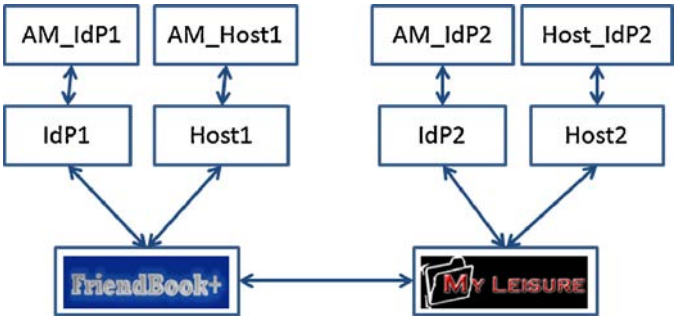


Fig. 8. Prototype architecture.

Firstly, it is assumed the existence of a pair of users, Alice and Bob, Alice enrolled in FriendBook+and Bob enrolled in MyLeisure. Then, regarding Fig. 8, Alice establishes her identity data and resources decryption keys in IdP1, encrypted resources in Host1, her access control policies in AM_IdP1 and AM_Host1 and, finally, her private key locally, in her personal computer, to perform decryptions. By contrast, Bob establishes his identity data in IdP2 and resources in Host2 and also uses a couple of AMs, AM_IdP2 and AM_Host2, to establish and manage access control policies. Afterwards, once Alice enrols in FriendBook+, IdP1 and Host1 are linked to it. Likewise, when Bob enrols in MyLeisure, he specifies where his identity data and resources are stored, that is, in IdP2 and Host2 respectively. The experimental evaluation verifies that Alice from FriendBook+is able to access to Bob identity data and resources in MyLeisure as Fig. 9 depicts.

Different technologies have been applied in the prototype development. J2EE and J2SE 1.6 has been used for implementing the pair

Fig. 9. Alice access to Bob's data.

of proposed WBSNs, FriendBook+and MyLeisure. Glassfish 3.0.1 has been applied to manage IdPs, Hosts and AMs and MySQL 5.2.27 to store all required data. Additionally, to measure network communications, Firebug 1.7.3 (a Firefox extension) has also been used. Besides, in respect to cryptographic algorithms, the scheme proposed in Section 7.1 is followed. In relation to symmetric cryptography, AES 128 is used to encrypt/decrypt resources (photos) and in respect to asymmetric cryptography RSA 2048 is the algorithm applied, assuming that each user owns a certificate and a private key of length 2048 bytes. On the other hand, photos managed in this implementation have a size between 200 kb and 300 kb. This size is chosen as an upper limit as the average size considering (Lunt et al., 2006).

10.2. Experimental results

The temporal workload of the protocol is analyzed in this section. It is measured the time spent while executing different protocol phases. More specifically, the access to personal identity data (profile), the access to a personal resource (photo), the access to a direct contact identity data (profile) and the access to a direct contact resource (photo) are analyzed. With this results, it is estimated and analyzed the workload of accessing an indirect contact identity data and an indirect contact resource. The study begins presenting, in Section 10.2.1, the prototype analysis results and concludes detailing, in Section 10.2.2, a comparison between the prototype and three successful WBSNs, Facebook, MySpace and LinkedIn.

10.2.1. Temporal workload

The total workload of performing any kind of access is measured as the cost of interchanging protocol messages until reached the requested data ($C_{dataAcquisition}$) multiplied by a parameter ψ (that corresponds to information not reused from previous requests) plus the cost of performing the required decryptions ($C_{dataDecryption}$), Eq. (1). Moreover, an analysis regarding possible values of ψ is performed by comparing the number of signatures carried out and the number of messages interchanged in the worst case (no elements are reused) and in the best case (all possible elements are reused) when a user logs in to a WBSN, a user accesses a direct contact's data and a user accesses an indirect contact's data (see Table 7). Considering that reusing is unachievable regarding the acquisition of the personal identity data because it is the first requested data, the performed analysis, presented in Table 8, shows that 68.75% of signatures and 51% of messages are reused, concluding that, on average, the maximum level of reuse is 59.87%. Consequently, three values of ψ are considered, ψ is 1 when not a single piece of data are reused, 0.70 when 50% of data are reused and 0.41 when all data are reused, that is 59.87%. The workload has been measured as the average of 10 executions and executions have been carried out without supposing the reuse of any element ($\psi = 1$).

$$C_{total} = C_{dataAccess} \cdot \psi + C_{dataDecryption} \quad (1)$$

Table 7
Analysing the reuse of data.

	Signatures	# Messages
Without reusing ($\psi = 1$)		
Login	4	24
FOAF file acquisition	2	12
Resource acquisition	2	12
Access to direct contact	16	50
FOAF file acquisition	8	25
Resource acquisition	8	25
Access to indirect contact	$16 \cdot (N+1)$	$50 \cdot (N+1)$
FOAF file acquisition	$8 \cdot (N+1)$	$25 \cdot (N+1)$
Resource acquisition	$8 \cdot (N+1)$	$25 \cdot (N+1)$
Maximum reuse		
Login	2	18
FOAF file acquisition	2	12
Resource acquisition	–	6
Access to direct contact	2	18
FOAF file acquisition	1	9
Resource acquisition	1	9
Access to indirect contact	$2 \cdot (N+1)$	$18 \cdot (N+1)$
FOAF file acquisition	$1 \cdot (N+1)$	$9 \cdot (N+1)$
Resource acquisition	$1 \cdot (N+1)$	$9 \cdot (N+1)$

N, (# of users in the relationship) – 1, $N > 1$; –, an element/action not required.

According to these features, plot presented in Fig. 10 depicts the workload of accessing to the profile and to a photo of a user registered in WBSN1 (FriendBook+) and to the profile and to a photo of a direct contact enrolled in WBSN2 (MyLeisure). Furthermore, in order to distinguish $C_{dataAcquisition}$ and $C_{dataDecryption}$, the workload regarding these individual costs in the worst case, that is, $\psi = 1$, is also presented. It is identified that $C_{dataAcquisition}$ implies a high workload while, $C_{dataDecryption}$ is rather small. Although cryptographic operations depend on the applied algorithm, the decryption scheme draws satisfactory results. Decryptions take 86.83 ms on average, 83.83 ms for profiles and 89.83 ms for photos. Recalling that profiles are encrypted through an asymmetric algorithm and photos through a symmetric one and the fact that asymmetric algorithms are slower than symmetric algorithms, results show that photos are bigger in size than profiles and then, the workload is rather similar.

Analysing the same features as in the previous plot, except for the access to the profile of a user registered in WBSN1 (because reuse is not possible), Fig. 11 presents workloads in regard to different ψ values. It is remarkable that to achieve successful results, reuse is a matter of concern. Besides, as expected, interoperability between WBSN1 and WBSN2 increases the workload. The difference between accessing to a particular data in WBSN1 and accessing to WBSN2 is 1191.90 ms when $\psi = 1$, 834.33 ms when $\psi = 0.70$ and 488.68 ms when $\psi = 0.41$.

On the other hand, the establishment of indirect relationships is a challenging goal achieved in eU+F but not implemented in the prototype. Nonetheless, assuming that an indirect relationship is composed of direct ones, the workload is estimated as the cost of data acquisition multiplied by ψ and by the length of indirect relationships (n) plus the cost of data decryption (see Eq. (2)). Therefore,

Table 8
Analysing the reuse of data in eU+F.

	Worst case	Best case	% reuse	Avg. reuse %		$_{maxReuse}$
# Messages						
Long-in	24	18	25	51	59.87	0.41 (1–59%)
Access direct contact	50	18	64			
Access indirect contact	50	18	64			
Signatures						
Long-in	4	2	50	68.75		
Access direct contact	16	2	87.5			
Access indirect contact	16	2	87.5			

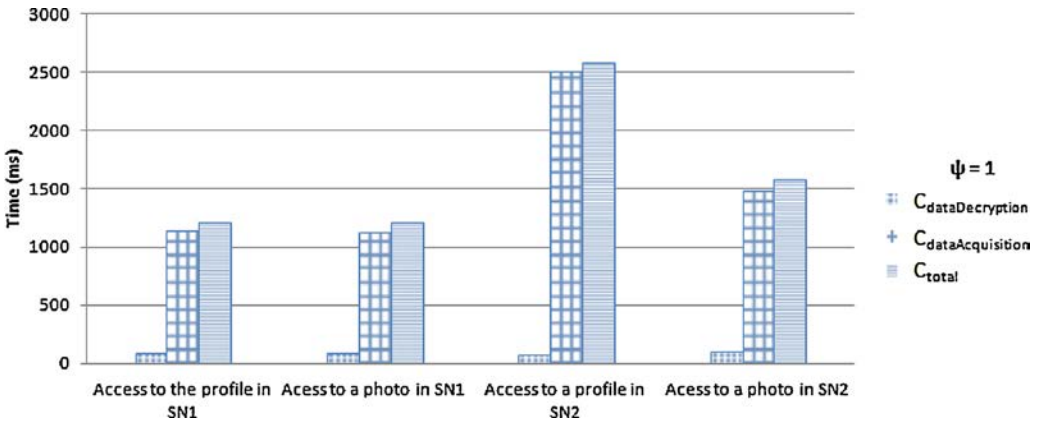


Fig. 10. Temporal costs comparison.

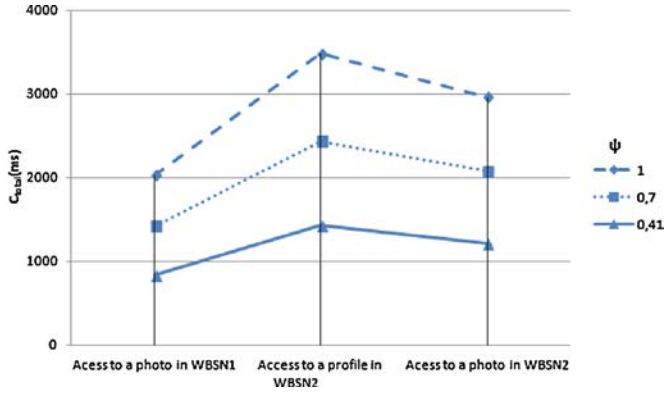


Fig. 11. General temporal workload.

plots presented in Fig. 12(a) and (b) show the estimated workload in respect to different values of ψ and n, given that n is bounded to 6 due to theoretical studies pointed out in Section 6.1.3.

$$C_{total} = C_{dataAccess} \cdot \psi \cdot n + C_{dataDecryption} \quad (2)$$

From the analysis it is identified that according to the longest indirect relationship ($n = 6$) in the worst case, $\psi = 1$, about 20,851 ms are needed to access a chosen profile and about 17,744 ms to a chosen photo. On the contrary, in the best case, $\psi = 0.41$ for $n = 6$, about 8,549 ms and 7,275 ms are taken to access a profile and to a photo respectively. Nevertheless, the reuse of data, for instance a user's credential, is highly probable and the average workload can be taken as a representative measure. In particular, for $\psi = 0.7$, workload is 2,432 ms for $n = 1$ and 14,596 ms for $n = 6$ to access a profile and 4,140 ms for $n = 1$ and 12,421 ms s for $n = 6$ to access a photo.

10.2.2. Comparison with other WBSNs

Three of the most currently successful WBSNs, Facebook, MySpace and LinkedIn, have been chosen to compare their workload with the experimental prototype (FriendBook+and MyLeisure). Firstly, in each WBSN, Facebook, MySpace and LinkedIn, a pair of accounts has been opened. Then, a set of six photos has been uploaded to the four opened accounts (three photos per account). In order to reach comparative results, the set of six photos is the same as the one managed in FriendBook+and MyLeisure. Nonetheless, LinkedIn is a well known WBSN but not focused on photo sharing and then, photos have been uploaded as profile photos and just one of them remains visible. Afterwards, using FireBug, the time to access to the personal profile, to a personal photo, to a direct contact's profile and to a direct contact's photo in all WBSNs is measured. Note that although the prototype does not implement the authentication, that is included in the login phase, the prototype workload of accessing to the personal profile of a user can be compared with the one measured in Facebook, MySpace and LinkedIn because authentication techniques of WBSNs like these are based on passwords and thus, the workload of such simple technique can be disregarded. According to Eq. (1), as in current WBSNs cryptographic techniques are not applied, the analyzed workload is bounded to $C_{dataAcquisition}$.

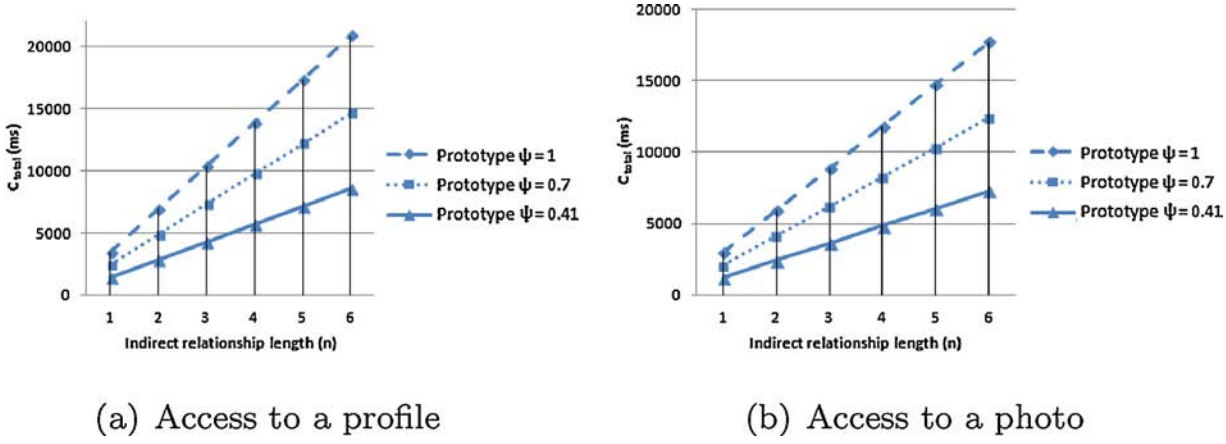


Fig. 12. Estimation of temporal workload for indirect relationships.

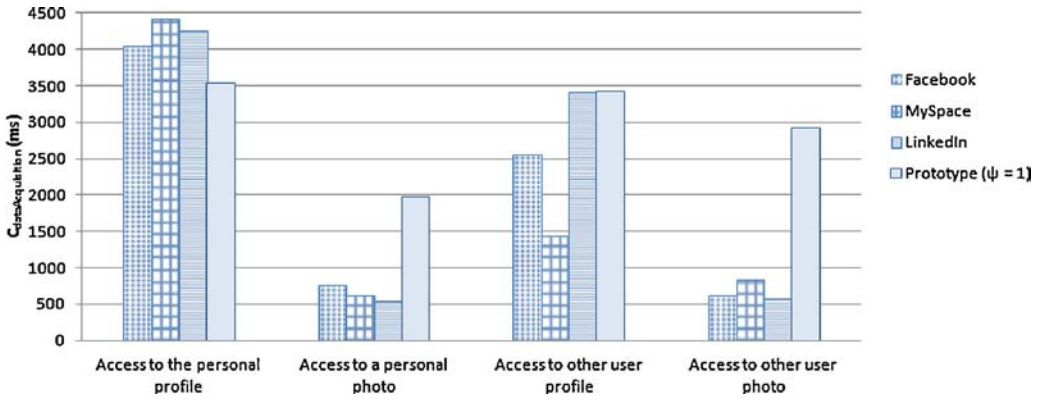


Fig. 13. Prototype, Facebook, MySpace and LinkedIn total cost comparison.

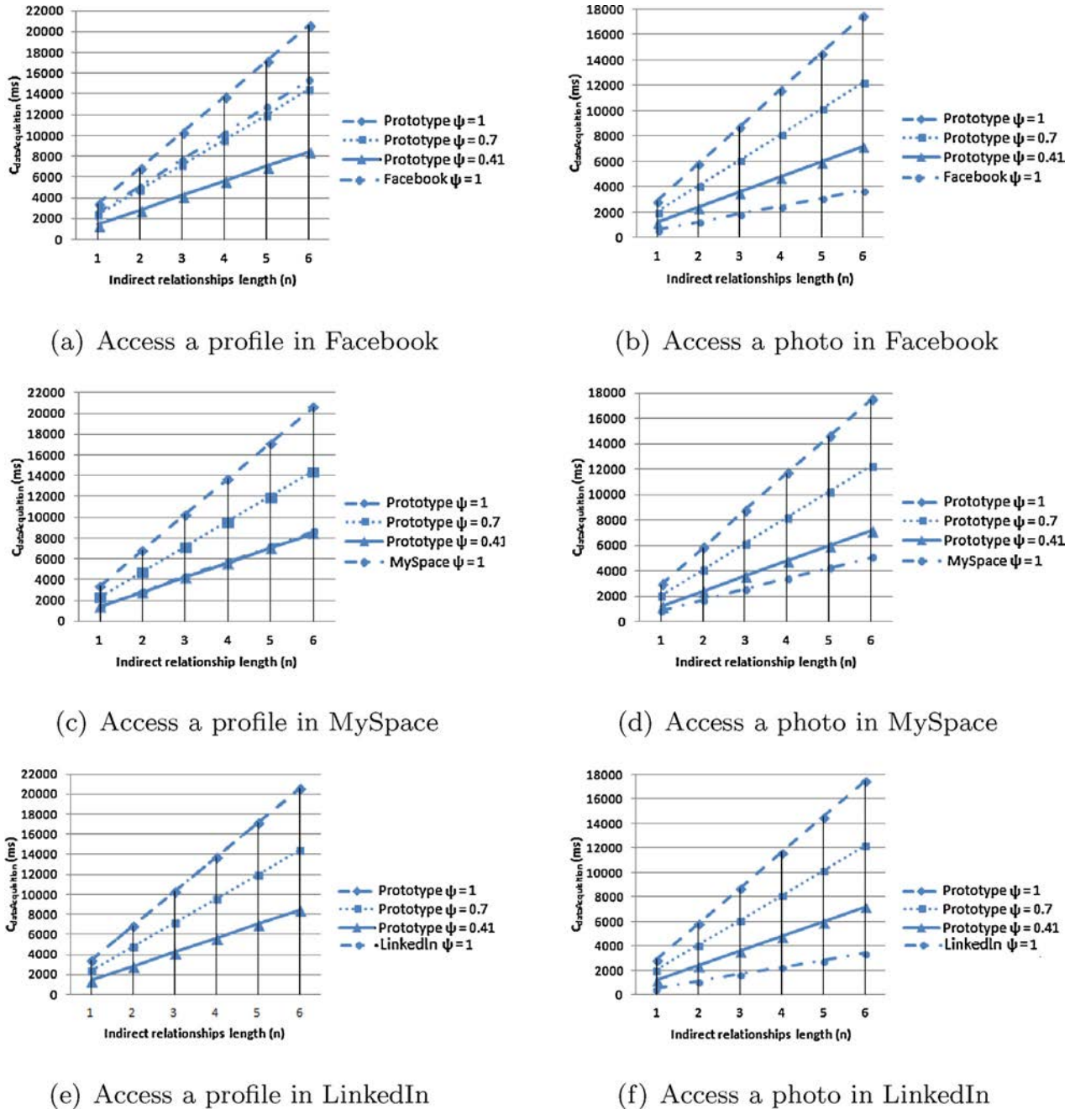


Fig. 14. Facebook, MySpace and LinkedIn indirect relationships comparison.

The comparison is presented in Fig. 13. It compares temporal costs for accessing to the profile of a user when he logins in a WBSN, to one of his photos and to the profile and to a photo of a direct contact. These costs are respectively 3,529 ms, 1,878 ms, 3,432 ms and 2,913 ms for the prototype, whereas they are 4,423 ms, 626 ms, 1,438 ms and 842 ms in the case of MySpace, 4,052 ms, 766 ms, 2,556 ms and 624 ms in the case of Fakebook and 4,248 ms, 541 ms, 3,422 ms and 571 ms in the case of LinkedIn. In general, Facebook, MySpace and LinkedIn follow a similar pattern in all cases. All these WBSNs produce higher workload than the prototype accessing to the personal profile and lower in the remaining cases. In this regard some findings are highlighted. First, profiles are richer in Facebook, MySpace and LinkedIn than in the prototype and then, higher TW is expected. In particular, personal profiles are specially richer in MySpace, e.g. including videos, and despite the fact that its implementation and architecture are not available to the public, rich profiles can affect the TW. Second, reaching interoperability through the decentralization of identity data, resources and access control policies is a challenging issue which expectedly produces a workload increase. Then, the workload accessing to data of other user (profile or photo) is higher in the prototype than in any other compared WBSN. Surprisingly, the TW accessing to the profile of other user in LinkedIn is really close to the prototype. Nonetheless, since the real implementation and architecture of LinkedIn are not available to the public, conclusive results cannot be obtained. Besides, regarding the access to photos, either personal or of other user, the prototype's TW is the highest one. It can be caused by the amount of messages interchanged in the protocol. In any case, as explained below, the prototype is far from being developed by powerful software and deployed on optimized hardware mechanisms, in contrast to real WBSNs like the ones analyzed, which are developed by huge companies.

On the other hand, the relevance of indirect relationship management requires its analysis. First of all, to establish comparable parameters and even not being currently possible, it is assumed that Facebook, MySpace and LinkedIn allow the establishment of indirect relationships of a maximum length of six. Then, given Eq. (2), the workload is calculated considering $C_{dataAcquisition}$ multiplied by each relationship length. Besides, note that Facebook, MySpace and LinkedIn, as far as we know, do not reuse data because elements like tokens or claims are not used. Concerning this feature, Fig. 14 presents achieved results. On the whole, it is remarkable that Facebook indirect relationship workload is similar to the developed prototype accessing to a profile when 50% of elements are reused ($\psi = 0.7$) and lower than the prototype when accessing to a photo. Furthermore, results show that, when accessing to a profile MySpace workload is close to $\psi = 0.41$ in the prototype and it is lower than the prototype accessing to a photo when all possible elements are reused. In relation to LinkedIn, accessing to a profile is comparable with the prototype when not reusing ($\psi = 1$) and, by contrast, accessing to a photo in LinkedIn is lower than in the prototype although all possible elements are reused.

Finally, there are a couple of points to highlight against and in favour of the developed prototype. On the negative side, in the executions performed for this experimental evaluation, entities managed (IdPs, Host, ...) run in a local network and it is possible that $C_{dataAcquisition}$ was higher in a real environment. On the negative side, contrary to the prototype, it is presumable that big companies, which develop WBSNs like Facebook and MySpace and LinkedIn, own robust and efficient infrastructures and mechanisms, for instance cache servers, which help to speed users' requests, thereby achieving successful $C_{dataAcquisition}$ times. In conclusion, from the authors' point of view, taking into account the challenge of dealing with indirect relationships in an interoperable environment, the workload of eU+F can be considered reasonable.

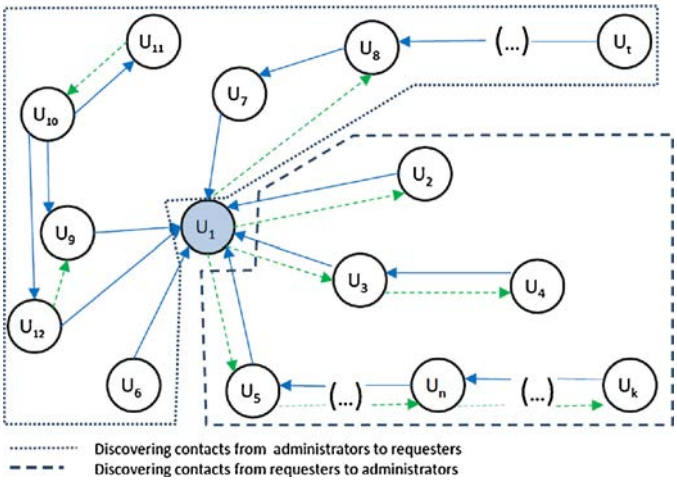


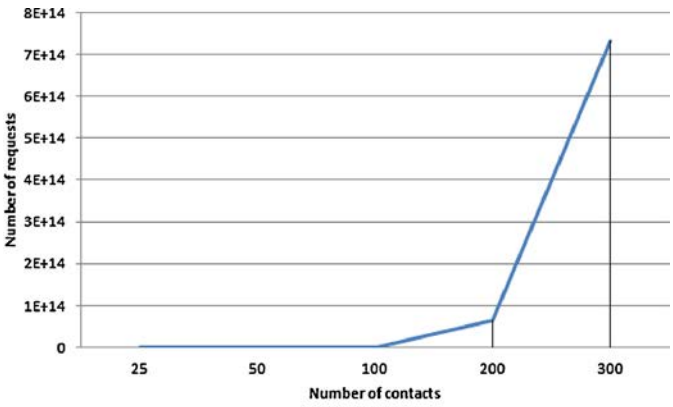
Fig. 15. Managing all types of relationships.

11. Discussion

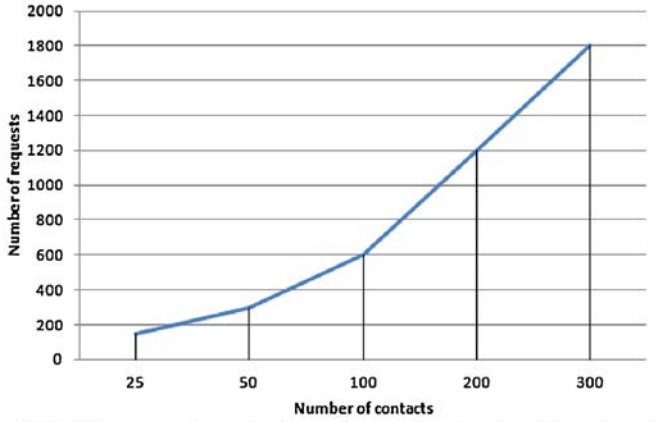
eU+F can be a more powerful approach. It can manage all kind of relationships without being restricted to the ones where the administrator is discovered from the requester, that is the approach currently taken. The management of this new set of relationships will be based on discovering contacts from the administrators to the requesters (contrary to the process currently performed in eU+F). Both approaches are depicted in Fig. 15. Therefore, with the new approach, the access control management will be carried out following the same direction as the contacts discovery. The new approach will work as follows.

Firstly, the requester specifies the administrator's email and a WBSN in which the administrator is supposed to be registered in and, subsequently, the procedure starts and it is described through the following examples. On the one hand, assuming that U_1 wants to access to U_6 (Fig. 15), the first step is equivalent to *User accesses to data of a direct contact*, described in Section 6.2.4, the access is granted if U_1 is in U_6's FOAF file and access control policies are satisfied. On the other hand, in a more challenging situation, U_1 wants to access to U_{10}'s data. In this case, the process starts requesting to each U_{10}'s contact the existence of a relationship with U_1. Thus, as U_9, U_{11} and U_{12} are the U_{10}'s contacts, the request is sent to IdP_U_9, IdP_U_{11} and IdP_U_{12}. Afterwards, if any contact has a relationship with U_1, the process is recursively repeated. In this example two users, U_9 and U_{12}, have a relationship with U_1 and due to that fact a pair of proofs are obtained, one that certifies U_{10}–U_1 through U_9 and another one that certifies U_{10}–U_1 through U_{12}. Finally, if when using some of the obtained proofs, access control policies are satisfied, access to U_{10}'s data are granted to U_1.

Nonetheless as it can be devised form the previous example, this process can be computationally hard. Firstly, it is possible that not a solution will be reached because a pair of users may not be connected. Secondly, if the network graph is significantly big, although relationship depth is bounded to six (Section 6.1.3), the number of requests to obtain a proof of the existence of a relationship may be unmanageable. For instance, supposing that a user wants to access to another that is distanced six jumps and also assuming a WBSN in which each user has n contacts on average, in the worst case the number of requests would be $\sum_{i=1}^{6} n^i$. By contrast, in the best case, the number of requests would be limited to $n \cdot 6$. More specifically, results of the number of requests regarding the worst and best case are depicted in Fig. 16(a) and (b) and it can be highlighted the high cost involved in discovering contacts.



(a) Discovering indirect contacts in the worst case



(b) Discovering indirect contacts in the best case

Fig. 16. Discovering indirect relationships in a powerful approach.

Despite highlighted drawbacks, this protocol could be extremely powerful because the whole network (independently of WBSNs where users are registered in) could be analyzed and even unknown relationships could be identified. Indeed, it is possible the identification of different relationships to reach a certain user and consequently, the possibilities of getting access to requested data increase. For instance, an administrator establishes that his data are only accessible to users with whom the established relationship is indirect with a maximum length of four. According to this policy, the more relationships between a requester and the administrator are found, the more possibilities of satisfying the access control policy there are.

12. Conclusions and open research issues

WBSNs are demanding developments and their research and improvement are key issues for the research community. In this regard, this proposal presents a protocol, called eU+F, to achieve interoperability and indirect relationships between WBSNs as well as the protection of data from WBSNs providers. The protocol is carefully detailed, describing goals, involved entities and all kind of interchanged messages. The protocol is also evaluated against the requirements and its performance is analyzed theoretically. Moreover, a eU+F prototype composed of a pair of WBSNs, Friendbook+and MyLeisure, has been developed. It shows the feasibility of implementing the protocol and its workload in comparison with some successful WBSNs, Facebook, MySpace and LinkedIn, is analyzed. Results drawn from the evaluation point out that eU+F can be considered an acceptable and challenging approach that, even supposing, in general, a workload increase in respect to current WBSNs, satisfactorily attains all established requirements.

This proposal can be extended in several ways. On the one hand, managing data exposure minimization through cryptography requires the analysis and selection of the most efficient cryptographic algorithm, being indispensable a comparative study of multiple algorithms. Likewise, the improvement of the proposal efficiency using lightweight cryptography (Eisenbarth and Kumar, 2007) or the application of the principle of asymmetry (Jiang et al., 2002) has to be also analyzed. Furthermore, the specification of constraints and rules to specify what it is considered a trusted IdPs and AMs is a future open issue. The idea is similar to the one proposed by Kang et al. (2011). They present the creation of *guardians*, people with a new profession, to protect personal data and it includes a detailed description of legal relations between *guardians* and clients. Other relevant matter is that currently, the protocol aborts if a particular AM, IdP or WBSN is not considered trusted (Section 6.2), being desirable a dynamic specification of trusted entities, for instance, requesting the appropriate administrator about the consideration of a new entity as a trusted one. Finally, a further step is to work towards the protocol improvement to reach a complete protection of users privacy by preventing WBSNs from inferring users relationships. In particular, regarding the access to data of an indirect contact, as pointed out in Section 6.2.4, a WBSN has to provide a proof of the existence of a relationship between a pair of users, being this proof noticed by the recipient WBSN. Due to that fact, after multiple executions of eU+F, WBSNs may infer the social network structure.

Appendix A. Technical details

In this Appendix messages managed in eU+F are detailed. Table A.1 presents the size in bits/bytes of each applied operation, element and structure. Furthermore, Tables A.2 and A.3 describe the size in bits/bytes of each eU+F message. Notice that structures which have not got a concrete size are limited by symbols ⟨ and ⟩ and they suppose an additional pair of bytes.

Table A.1
eU+F messages content technical specification.

	Bits	Bytes
Operations		
Signature (RSA-2048 + SHA1)	160	20
Encryption (RSA-2048)	$long_{variable}$	$long_{variable}$
Elements		
Token value	43	≈6
Token expiration time	13	≈2
File identifier	39	≈5
Ticket	39	≈5
User identifier	160	20
Date and time	32	4
Certificate serial number (Max.)	40	5
(Min.)	4	≈1
Redirection url (Max.)	16,384	2,048
(Min.)	4	≈1
Structures		
Relationship *X*–*Y*	451	≈57
Accreditation	272	34
Data request (Max.)	3,624	453
(Min.)	256	16
Data response (Max.)	44,704	5,588
(Min.)	264	33

Table A.2
eU+F messages: a technical specification (part I).

Id	Name	Content	Bits (b)	Bytes (B)	Total size Bytes
1	Token request	Ticket	39	≈5	
		<WBSN$_R$-$Cert\_Serial\_Number$> ($Max.$)	56	7	
		(Min.)	20	≈3	36 (Max.)/ 32 (Min.)
		Date_time$_{WBSN_R signature}$	32	4	
		$S_{k_{WBSN_R\text{-}Cert}}$(Complete Message)	160	20	
2	Token request redirection	Ticket	39	≈5	
		< AM_location > (Max.)	16400	2050	2055 (Max.)/ 8 (Min.)
		(Min.)	20	≈3	
3	Token response redirection	Ticket	39	≈5	
		Token value	43	≈6	
		Expired-in	13	≈2	
		<AM$_A$-$Cert\_Serial\_Number$> ($Max.$)	56	7	44 (Max.)/ 40 (Min.)
		(Min.)	20	≈3	
		Date_time$_{AM_A signature}$	32	4	
		$S_{k_{AM_A\text{-}Cert}}$(Complete Message)	160	20	
4	Token response	Token response redirection (Max.)	352	44	
		(Min.)	320	40	
		<WBSN$_R$-$Cert\_Serial\_Number$> ($Max.$)	56	7	75 (Max.)/ 67 (Min.)
		(Min.)	20	≈3	
		Date_time$_{WBSN_R signature}$	32	4	
		$S_{k_{WBSN_R\text{-}Cert}}$(Complete Message)	160	20	
5	File request	R_Id	160	20	
		A_Id	160	20	
		File_Id	39	≈5	
		<WBSN$_R$-$Cert\_Serial\_Number$> ($Max.$)	56	7	76 (Max.)/ 72 (Min.)
		(Min.)	20	≈3	
		Date_time$_{WBSN_R signature}$	32	4	
		$S_{k_{WBSN_R\text{-}Cert}}$(Complete Message)	160	20	
6	File indirect request	R_Id	160	20	
		A_Id	160	20	
		File_Id	39	≈5	
		<WBSN$_R$-$Cert\_Serial\_Number$> ($Max.$)	56	7	76 (Max.)/ 72 (Min.)
		(Min.)	20	≈3	
		Date_time$_{WBSN_R signature}$	32	4	
		$S_{k_{WBSN_R\text{-}Cert}}$(Complete Message)	160	20	
7	File response	R_Id	160	20	
		A_Id	160	20	40+E(file)
		E_{k_R}(file)			
8	Claims request	R_Id	160	20	
		A_Id	160	20	
		< $E_{k_{CertIdP_R}}$(Data R request) > (Max.)	3623	453	
		(Min.)	256	16	524 (Max.)/ 83 (Min.)
		<AM$_A$-$Cert\_Serial\_Number$> ($Max.$)	56	7	
		(Min.)	20	≈3	
		Date_time$_{AM_A signature}$	32	4	
		$S_{k_{CertAM_A}}$(Complete Message)	160	20	
9	Claims structures response	R_Id	160	20	
		A_Id	160	20	
		Accreditation R	272	34	
		< $E_{k_{CertAM_A}}$(Data R response) > (Max.)	44704	5588	
		(Min.)	264	33	5693 (Max.)/ 134 (Min.)
		<IdP$_R$-$Cert\_Serial\_Number$> ($Max.$)	56	7	
		(Min.)	20	≈3	
		Date_time$_{IdP_R signature}$	32	4	
		$S_{k_{IdP_R}}$(Complete Message)	160	20	
10	Claims response	Claims structures response (Max.)	45536	5692	
		(Min.)	1064	133	
		Relationship R_A$_1$	451	≈57	
		<IdP$_A$-$Cert\_Serial\_Number$> ($Max.$)	56	7	
		(Min.)	20	≈3	
		Date_time$_{IdP_A signature}$	32	4	5811 (Max.)/ 244 (Min.)
		$S_{k_{IdP_A}}$(Relationship R_A$_1$)	160	20	
		<WBSN$_R$-$Cert\_Serial\_Number$> ($Max.$)	56	7	
		(Min.)	20	≈3	
		Date_time$_{WBSN_R signature}$	32	4	
		$S_{k_{WBSN_R\text{-}Cert}}$(Complete Message)	160	20	

Table A.3
eU+F messages: a technical specification (part II).

Id	Name	Content	Bits	Bytes	Total size Bytes
11	Certify direct relationship	R_Id	160	20	193 (Max.)/ 185 (Min.)
		A_Id	160	20	
		Accreditation R	272	34	
		<IdP$_R$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Date_time$_{IdP_R signature}$	32	4	
		$S_{k_{IdP_R}}$(Accreditation R)	160	20	
		Relationship A_R	451	≈57	
		<WBSN$_R$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Date_time$_{WBSN_R signature}$	32	4	
		$S_{k_{WBSN_R\text{-}Cert}}$(Complete message)	160	20	
12	Certify indirect relationship	R_Id	160	20	281 (Max.)/ 266 (Min.)
		A$_X$_Id	160	20	
		Accreditation R	272	34	
		<IdP$_R$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Date_time$_{IdP_R signature}$	32	4	
		$S_{k_{IdP_R}}$(Accreditation R)	160	20	
		Relationship A$_i$_R	451	≈57	
		<IdP$_{A_i}$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Date_time$_{IdP_{A_i} signature}$	32	4	
		$S_{k_{IdP_{A_i}}}$ (Relationship A$_i$_R)	160	20	
		Relationship R_A$_1$	451	≈57	
		<WBSN$_A$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Date_time$_{WBSN_A signature}$	32	4	
		$S_{k_{WBSN_A\text{-}Cert}}$(Complete message)	160	20	
13	Relationship certified	R_Id	160	20	132 (Max.)/ 128 (Min.)
		A_Id	160	20	
		Relationship A_R	451	≈57	
		<IdP$_A$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Date_time$_{IdP_A signature}$	32	4	
		$S_{k_{IdP_A}}$(Relationship R_A$_1$)	160	20	
14	Simple token request	Ticket	39	≈5	5
15	Simple token response	Ticket	39	≈5	44 (Max.)/ 40 (Min.)
		Token value	43	≈6	
		Expired-in	13	≈2	
		<AM$_A$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Date_time$_{AM_A signature}$	32	4	
		$S_{k_{AM_A\text{-}Cert}}$(Complete Message)	160	20	
16	Simple file request	R_Id	160	20	25
		File_Id	39	≈5	
17	Simple file response	R_Id	160	20	20+E(file)
		E_{k_R}(file)			
18	Simple claim request	R_Id	160	20	40
		A_Id	160	20	
19	Simple claim response	R_Id	160	20	85 (Max.)/ 81 (Min.)
		<IdP$_R$_$Cert_Serial_Number$ > $(Max.)$	56	7	
		(Min.)	20	≈3	
		Accreditation R	272	34	
		Date_time$_{IdP_R signature}$	32	4	
		$S_{k_{IdP_R}}$(Accreditation R)	160	20	
20	Token validation	Ticket	39	≈5	31
		Token value	43	≈6	

References

Ackermann, M., Ludwig, B., Hymon, K., Wilhelm, K., 2009. Helloworld: an open source, distributed and secure social network. In: W3C Workshop on the Future of Social Networking.

Acquisti, A., Gross, R., 2006. Imagined communities: awareness, information sharing, and privacy on the Facebook. In: Privacy Enhancing Technologies. Springer, Berlin/Heidelberg, pp. 36–58.

Aiello, L.M., Ruffo, G., 2012. Lotusnet: tunable privacy for distributed online social network services. Comput. Commun. 35, 75–88.

Anderson, R., 2001. Security Engineering: A Guide to Building Dependable Distributed Systems.

Baden, R., Bender, A., Spring, N., Bhattacharjee, B., Starin, D., 2009. Persona: an online social network with user-defined privacy. SIGCOMM Comput. Commun. Rev. 39, 135–146.

Boneh, D., Franklin, M., 2001. Identity-based encryption from the Weil pairing. In: Advances in Cryptology—CRYPTO 2001. Springer, Berlin Heidelberg, pp. 213–229.

Borcea-Pfitzmann, K., Pfitzmann, A., Berg, M., 2011. Privacy 3.0 := data minimization + user control + contextual integrity. Inform. Technol. 53, 34–40.

Carminati, B., Ferrari, E., Perego, A.,2006. Rule-based access control for social networks. In: Proceedings of the OTM 2006 Workshops (On the Move to Meaningful Internet Systems). Springer, pp. 1734–1744.

Carminati, B., Ferrari, E., Perego, A.,2007. Private relationships in social networks. In: Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering Workshop. IEEE Computer Society, pp. 163–171.

Carminati, B., Ferrari, E., Heatherly, R., Kantarcioglu, M., Thuraisingham, B.,2009. A semantic web based framework for social network access control. In: Proceedings of the 14th ACM Symposium on Access Control Models and Technologies. ACM, pp. 177–186.

Ciriani, V., De Capitani, S., Foresti, S., Jajodia, S., Paraboschi, S., Samarati, P., 2011. Selective data outsourcing for enforcing privacy. J. Comput. Secur. 19, 531–566.

Conti, M., Hasani, A., Crispo, B., 2011. Virtual private social networks. Proceedings of the First ACM Conference on Data and Application Security and Privacy, 39–50.

di Vimercati, S.D.C., Foresti, S., Jajodia, S., Paraboschi, S., Samarati, P.,2007. A data outsourcing architecture combining cryptography and access control. In: Proceedings of the 2007 ACM Workshop on Computer Security Architecture. ACM, pp. 63–69.

Eisenbarth, T., Kumar, S., 2007. A survey of lightweight-cryptography implementations. Des. Test Comput. IEEE 24, 522–533.
FOAF Team, 2000. FOAF Project. http://www.foaf-project.org/
Frikken, K.B., Srinivas, P.,2009. Key allocation schemes for private social networks. In: Proceedings of the 8th ACM Workshop on Privacy in the Electronic Society. ACM, pp. 11–20.
González-Manzano, L., González-Tablas, A., de Fuentes, J., 2012. U+F Social Network Protocol: achieving interoperability and reusability between web based social networks. In: IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications.
González-Manzano, L., González-Tablas, A., de Fuentes, J.M., Ribagorda, A.,2013. User-managed access control in web based social networks. In: Security and Privacy Preserving in Social Networks. Springer.
González-Manzano, L., González-Tablas, A.I., de Fuentes, J.M., Ribagorda, A., 2014. SoNeUCON$_{ABC}$, an expressive usage control model for web-based social networks. Comput. Secur. 43, 159–187.
González-Tablas, A.I., Alam, M., Hoffmann, M., 2010. An architecture for user-managed location sharing in the future internet of services. In: The 4th International Workshop on Trustworthy Internet of People, Things & Services, Co-located with the Internet of Things 2010 Conference.
Google Team, 2012. Google: Privacy Policy. http://www.google.com/intl/en/policies/privacy/
Graffi, K., Gross, C., Mukherjee, P., Kovacevic, A., Steinmetz, R., 2010. LifeSocial.KOM: a P2P-based platform for secure online social networks. In: 2010 IEEE Tenth International Conference on Peer-to-Peer Computing (P2P), pp. 1–2.
Guha, S., Tang, K., Francis, P.,2008. NOYB: privacy in online social networks. In: Proceedings of the First Workshop on Online Social Networks. ACM, pp. 49–54.
Hardjono, T., 2012. User-Managed Access (UMA) Core Protocol, draft-hardjono-oauth-umacore-05c.
Jammalamadaka, R.C., Gamboni, R., Mehrotra, S., Seamons, K., Venkatasubramanian, N.,2008. iDataGuard: an interoperable security middleware for untrusted internet data storage. In: Proceedings of the ACM/IFIP/USENIX Middleware '08 Conference Companion. ACM, pp. 36–41.
Jiang, X., Hong, J.I., Landay, J.A.,2002. Approximate information flows: socially-based modeling of privacy in ubiquitous computing. In: Ubicomp 2002: Ubiquitous Computing. Springer, pp. 176–193.
Joe, M., 2009. Identity-based Cryptography. IOS Press.
Kang, J., Shilton, K., Estrin, D., Burke, J., Hansen, M., 2011. Self-surveillance privacy. Datenschutz Datensicherheit – DuD 35, 624–628.
Kantara Members, 2009. Kantara Initiative. http://kantarainitiative.org/
Kourtellis, N., Finnis, J., Anderson, P., Blackburn, J., Borcea, C., Iamnitchi, A., 2010. Prometheus: user-controlled P2P social data management for socially-aware applications. IFIP – Int. Feder. Inform. Process., 212–231.
Lucas, M.M., Borisov, N.,2008. FlyByNight: mitigating the privacy risks of social networking. In: Proceedings of the 7th ACM Workshop on Privacy in the Electronic Society. ACM, pp. 1–8.
Lunt, C., Abrams, J., Sanchez, S., 2006. Method of Inducing Content Uploads in a Social Network. United States Patent 7,117,254. http://www.google.com/patents/US7117254 (last accessed March 2014).
Machulak, M., Maler, E., Catalano, D., van Moorsel, A., 2010. User-managed access to web resources. In: Proceedings of the 6th ACM Workshop on Digital Identity Management, pp. 35–44.
Maurer, U.M., Yacobi, Y., 1996. A non-interactive public-key distribution system. Des. Codes Cryptogr. 9, 305–316.
Riesner, M., Pernul, G., 2012. Provider-independent online social identity management-enhancing privacy consistently across multiple social networking sites. In: 2012 45th Hawaii International Conference on System Science (HICSS), pp. 800–809.
Salomaa, A., 1996. Public-key Cryptography, vol. 23. Springer.
Saltzer, J.H., Schroeder, M.D., 1975. The protection of information in computer systems. Protect. Inform. Comput. Syst., 1278–1308.
Seong, S.W., Seo, J., Nasielski, M., Sengupta, D., Hangal, S., Teh, S.K., Chu, R., Dodson, B., Lam, M.S., 2010. PrPl: A Decentralized Social Networking Infrastructure., pp. 8:1–8:8.
Tootoonchian, A., Saroiu, S., Ganjali, Y., Wolman, A., 2009. Lockr: Better Privacy for Social Networks., pp. 169–180.
Yeung, C.M.A., Liccardi, I., Lu, K., Seneviratne, O., Berners-Lee, T., 2009. Decentralization: the future of online social networking. In: W3C Workshop on the Future of Social Networking Position Papers.

Lorena González-Manzano is a PhD student working in the Computer Security Lab at the University Carlos III of Madrid, Spain. After finishing her degree of Computer Engineering in October 2010 and a Master in Computer Science and Technology in February 2012 at the University Carlos III of Madrid, she started working as a PhD student in this university. Particularly, she is working on privacy in social networks, as well as, information security issues. Indeed, she has published several papers in national and international conferences and journals and she is also involved in national R+D projects.

Ana Isabel González-Tablas is associate professor in the Computer Science and Engineering Department at University Carlos III of Madrid. She is Telecommunications Engineering by the Polytechnic University of Madrid, Spain, since 1999 and received her PhD degree in Computer Science from University Carlos III of Madrid, Spain, in 2005. Her main research interests are security and privacy for Intelligent Transportation Systems and Location Based Services. She has participated, and currently participates, in several national and European research projects. She has published numerous articles in national and international journals and conferences.

José María de Fuentes is teaching assistant in the Computer Science and Engineering Department at University Carlos III of Madrid, Spain. He is Computer Scientist Engineer and PhD in Computer Science by the University Carlos III of Madrid. His main research interests are digital evidences management, non-repudiation in vehicular environments, as well as security and privacy in social networks. He has published several articles in international conferences and journals. He is participating in several national R+D projects.

Arturo Ribagorda Garnacho is professor of the University Carlos III of Madrid, Spain. He is Telecommunications Engineer by the Polytechnic University of Madrid, Spain, and PhD in Computer Science by this university. His researches focus on security in information and communications technology. His has been involved in national and international projects, published multiple papers also in journals and being speaker in assorted conferences. Furthermore, he has written four books regarding security issues and he is assessor of multiple projects. Currently, he is the head of the Computer Security Lab Group at the University Carlos III of Madrid.