Anonymity and closely related terms in the Cyberspace: An analysis by example

Georgios Kambourakis

Info-Sec-Lab Laboratory of Information and Communications Systems Security, Department of Information and Communication Systems Engineering, University of the Aegean, Samos, Greece

Abstract

Anonymity is generally conceived to be an integral part of user's right to privacy. Without anonymity, many online activities would become prone to eavesdropping, making them potentially risky to use. This work highlights on the different aspects closely related to anonymity and argues that it is rather a multifaceted and contextual concept. To support this argumentation, the paper examines as a dual case study the ways anonymity is conceptualised in the case of two well-established but dissimilar protocols employed in the cyberspace on a wide-scale; that is, SIP and Kerberos ones. By surveying the research done for preserving anonymity (and privacy in general) in the context of the aforementioned protocols several useful observations emerge. Our aim is to contribute towards acquiring a comprehensive view of this particular research area, mainly by examining how anonymity is put to work in practice. As a result, the work at hand can also be used as a reference for anyone interested in grasping the diverse facets of this constantly developing research field.

Keywords: Anonymity, Identity protection, Privacy, Survey, SIP, Kerberos.

*Department of Information and Communication Systems Engineering, University of the Aegean, Samos GR-83200, Greece.

Email address: gkamb@aegean.gr (Georgios Kambourakis)

Feb., 2014

1. Introduction

Privacy concerns are constantly gaining more and more attention as the Internet grows in a second-by-second basis along with the importance of what people do online. It is therefore without a doubt that the provision of anonymity enables individuals to perform their online activities in comfort and privacy. In fact, anonymity has arisen as a valuable weapon in the battle against eavesdropping and other dangers lurking in the open Internet, including online identity theft, fraud, spam, and phishing. The very recent disclosure of the NSA PRISM surveillance program is indeed self-witnessing of how easily an interested party having enough resources is able to unleash a large scale eavesdrop on peoples' online communications (PRISM, 2013).

Primarily, anonymity has to do with identity protection, which in the cyberspace is usually achieved through some sort of pseudonymity. However, as it is discussed further down in the next section, anonymity is rather a multifaceted concept, and it needs to be dealt differently depending on the situation. For instance, tightly controlled environments, like that of a military network, may leave no room for anonymity, while others, such as a chat room, can at least ensure an acceptable level of anonymity to their users. Moreover, sometimes, anonymity may be very useful for Service Providers (SP) as well. For example, many providers would be interested in hiding data about which their most popular (accessed) service is. Having all the above in mind the work in (Cameron, 2006) poses a fundamental question and answers it appropriately: *"Why is it so hard to create an identity layer for the Internet? Mainly because there is little agreement on what it should be and how it should be run. This lack of agreement arises because digital identity is related to context, and the Internet, while being a single technical framework, is experienced through a thousand kinds of content in at least as many different contexts - all of which flourish on top of that underlying framework"..*

Where matters, anonymity can be imposed in several layers of the Internet model. Actually, one could agree that the lower the layer the stronger the level of anonymity. This means that enforcing anonymity at the application layer only, may be not enough for environments where a strong flavor of this service is desired. That is, while the identities (IDs) of the communicating parties may remain hidden, say, due to a pseudonymity scheme applied at the application layer, their IP addresses leak out in absence of a protection mechanism at the network layer. Anonymity also is closely related to ac-

countability. In the real world we are as a rule accountable for our actions. In cyberspace though we are not. In a plethora of cases this is a nice thing; in others, not so much. Namely, without accountability, it could be very hard, if not impossible, to deter and cope with the various forms of offensive user behavior that endanger the smooth operation of the network. Therefore, a fundamental question arises: is there a way to promote the positive values of anonymity while keeping an acceptable level of accountability at the same time?

So far, many works in the literature have been devoted to identity protection and anonymity in general. The majority of them are aiming in proposing some user identity protection scheme for a basic protocol or implementing novel solutions that consider anonymity in a "by design" fashion. This trend does not come as a surprise - as most solutions have been initially built by giving emphasis on functionality first rather on anonymity - and is sure to get bigger in the years to come especially for prime research topics. For instance, the authors in (Pereñiguez-Garcia et al., 2010) proposed a mechanism to offer user anonymity and untraceability for fast re-authentication processes in Extensible Authentication Protocol (EAP)-based (Aboba et al., 2004) next generation networks. Note that although EAP was initially conceived for network access authentication, its applicability to support other scenarios like application layer authentication is currently being evaluated (Winter and Salowey, 2013). In this respect, enabling anonymity in EAP-based networks, even as an opt-in, is sure to gain more and more attention in the near future.

During the last years we are also witnessing a movement towards the standardisation of anonymity-friendly solutions in the form of an RFC or Internet draft. A characteristic example of this situation is given in RFC 5636 (Park et al., 2009), which defines an architecture and protocols for offering privacy to users who request and use an X.509 certificate called Traceable Anonymous Certificate. The latter contains a pseudonym, but the ability to map such a certificate to the real user who solicited it is still retained. In the same context, (Simon et al., 2008) provides a special privacy extension that allows the peer's certificate to be sent within a TLS session supporting confidentiality. This tendency verifies the growing interest about the - many times - conflicting issues revolving around anonymity, and especially those that try to balance between the freedom to be anonymous and the proper tracking of digital assets and functioning of the network.

Our contribution: This paper attempts to examine and conceptualise the

various ways anonymity is (or can be) imposed in the cyberspace. This is mainly done by exploring the different facets it presents, its interplay with accountability, and its relation to the different layers of the Internet model. This effort is also backed-up by surveying, as a case study, the ways anonymity is considered in the context of some well-established protocols used in the cyberspace. This allows us to elaborate on the different aspects of anonymity depending on the situation at hand. Our aim is to help towards grasping a holistic view of this particular research area, mainly by examining the road so far. Hence, the current work can also be used as a reference to anyone interested in better understanding the different facets of this fast evolving area. It is also expected to foster research efforts to the development of full-fledged solutions that put emphasis mostly to the technological, but also to the standardization aspect.

The rest of the paper is structured as follows. The necessary background and definitions on anonymity are given in the next section. Section 3 details on the ways in which anonymity is defined and treated in the context of major protocols used in the cyberspace. This, when combined with the key points of section 2, allows for an analysis on the effectiveness and practicality of the solutions proposed so far, and reveals limitations and significant directions for future work. The last section draws a conclusion.

2. Anonymity: the quest for being nameless

This section puts together all the major pieces of the anonymity puzzle. Specifically, it provides definitions to terms related to anonymity and elaborates on the association between anonymity and closely to it aspects like that of accountability. This discussion serves as a base line for the analysis of the case studies provided in section 3.

2.1. Definitions and Interplay with other Terms

According to (Burkell, 2006), in the social science literature, the term anonymity includes three distinct aspects: *identity protection*, *action anonymity* and *visual anonymity*. First and foremost, identity protection corresponds to the situation where the subject remains unidentified. Note that in the online world the term *identity* pertains to the representation of an entity (a person in our case) in a specific context (application domain) and is usually related to a real world entity. Each entity is described by attributes attached to it (e.g., name, biological and social characteristics, competences,

location, personality, etc). Hence, the previous definition also refers to Personally Identifiable Information (PII) which according to (ITU-T, 2009) is defined as: *"the information pertaining to any living person which makes it possible to identify such individual (including the information capable of identifying a person when combined with other information even if the information does not clearly identify the person)."*. On the other hand, action anonymity has to do with the fact that individuals may feel accountable for their actions (or feel known by their actions) even in cases their real identity remains well-hidden. This explains the situation that sometimes the actions of a person may bespeak more about them than knowing their face or name. Lastly, visual anonymity is achieved when one's face goes unnoticed (e.g., when being disguised or wearing a mask). Therefore, this third aspect of anonymity is much more attainable and considered de facto in online interactions excluding of course situations where a person opts to give away their identity by, say, posting a photo of themselves in the Facebook.

The works in (Hansen et al., 2011; Pfitzmann and Hansen, 2010) also elaborate on the definition of the anonymity term, but this time, specifically for the cyberspace. *"Anonymity of a subject from an attacker's perspective means that the attacker cannot sufficiently identify the subject within a set of subjects, the anonymity set"* (Hansen et al., 2011). As it can be observed, this definition includes all the aforementioned three aspects spotted by (Burkell, 2006). Particularly, identity protection is usually imposed via some sort of *pseudonymity*, where *"a pseudonym is an identifier of a subject other than one of the subject's real names"* (Hansen et al., 2011). More specifically, a *person pseudonym* consists a substitute of the owner's real name applicable to multi-purposes. A *role pseudonym* is closely related to specific context, think for example a student registration number. On the other hand, a *transaction pseudonym* serves as an one-time identity and it is considered valid for a single transaction only.

Also, it can be argued that anonymity is closely related to *unlinkability* (*"...within a particular set of information, the attacker cannot distinguish whether a number of items of interest are related or not ..."* (Hansen et al., 2011)). In fact, the unlinkability property can be seen as a more advanced kind of anonymity since eavesdroppers are not only unable to infer the identity of a user but also to derive any useful relationship between the exchanged messages and the user itself. This also means that action anonymity as given in (Burkell, 2006) is certain to entail some sort of unlinkability. Therefore, pseudonymity realised through transaction pseudonyms provides the

strongest degree of identity protection, as messages - even those exchanged within the same session - are very difficult to be linked with one another. This is in contrary to person pseudonyms which provide the highest degree of linkability and the weakest degree of identity protection. A last point of interest here is the difference between the terms anonymity and pseudonymity. This actually refers to what kind of ID is used in the place of the real user ID. A completely anonymous scheme can utilise static strings like "anonymous@anonymous.invalid" (see section 3.1) or completely random strings. For a scheme based on pseudonymity the replacement ID is produced in some way from the real user's ID. Anonymous schemes are purely stateless, meaning that session state in a given transaction cannot be preserved. On the downside, in poorly designed pseudonymity schemes, say when the same pseudonym is used repeatedly, a user can be tracked down even if the correspondence between the real ID and the user ID is kept secret.

From the above it becomes clear that anonymity is a complex and multifaceted concept. Admittedly, this also holds true for privacy itself which is multidimensional, context-depended, and therefore hard to define (Hoepman, 2013). For example, as elaborated in (Kim, 2010), in an online forum or group, individuals might care little about action anonymity (their history of action) but surely yearn for remaining anonymous and unseen. This is especially true for people who are interested in bypassing censorship, persons with extreme political beliefs, stigmatised identities, journalists, and others. Nevertheless, stories like that of "Gaydar" (Johnson, 2009) where two MIT students discovered that by using Facebook friend links one could predict whether the person is homosexual, prove that true (full) anonymity is hard to be achieved. In particular, such threats to action anonymity reveal that online interactions leak sensitive information which in the short- or mid-term may lead to identifying a person. Putting it another way, who you really are can be divulged by who your friends and what your actions are. This is where unlinkability may be proved useful. Nevertheless, this is not to be taken for granted, as there is little room for unlinkability in, say, social networking sites. Also, it is not to be ignored that in the cyberspace, one's identification credentials, including IDs, IP addresses and others are usually *"recorded in databases, compared or collated with other data, and stored indefinitely for further uses"* (Cavoukian, 2006). Once more, this situation suggests that the emergence of a single anonymity solution - in terms of digital identity - to cover every need is rather unfeasible. For obtaining a more complete view on the subject the interested reader can also refer to the directive 95/46/EC of

the European Parliament on the protection of individuals with regard to the processing of personal data and on the free movement of such data (E.U., 1995).

2.2. Anonymity vs. Accountability

As already mentioned in section 1, anonymity is closely related to *accountability*, especially with reference to tightly controlled settings. For example, anonymity may be totally undesirable in a classified government or military network. Actually, as argued in (Wolff, 2013), there is an active debate on whether online anonymity protections are conflicting with robust accountability mechanisms. So, to which degree and to what of its aspects anonymity is desirable heavily depends on the particular case at hand. Therefore, contrary to the common belief that anonymity is by-default inconsistent with accountability (Davenport, 2002), several works argue in favor of the opposite. That is, accountability can exist even in cases of weak authentication (Wolff, 2013; Schneier, 2012). Moreover, as stated in (Wolff, 2013), this can be achieved through the implementation of a variety of context-specific accountability mechanisms at the application layer, rather than a single, uniform mechanism at the network layer.

In the same work the authors identify some ways (patterns) that are certain to become handy when a fair balancing between identity-protection and accountability is desired. Specifically, they observe that a virtual identity (e.g., a pseudonym) is closely bound to what the user invests into creating and using it. This observation is indeed critical and leads to the implementation of solutions that could possibly prevent a person from creating new usernames at will. For example, the user may need to spend some money in order to create a new identity in the system. Also, this identity may worth a lot to its owner who has devoted much time in increasing its reputation. Consequently, the suspension or permanent deletion of that identity, due to misbehavior, could have a considerable cost to its user. In other words, the more one spends on an identity the more it worth to them. The authors refer to this situation by the term *identity investment-privilege trade-off* and identify certain forms of privilege and investment used by real life applications to impose accountability to anonymous users. They also point out that this trade-off is useful both to the end-users, and application designers and operators towards setting the boundaries of accountability and anonymity preferences they believe to be more appropriate depending on the case.

Another way to offer anonymity, but also keep control of who is doing what, is to implement a *conditional anonymity* scheme. This in fact stems from the so-called *revocable privacy* defined in (Hoepman, 2013) as "*A system implements revocable privacy if the architecture of the system guarantees that personal data is revealed only if a predefined rule has been violated.* Typically, such a solution requires from the end-users to entrust some type of real credentials to a Trusted Third Party (TTP) or the service operator upon their registration. In this way, it offers a King Solomon solution to the problem; your real identity remains hidden as long as you are willing to abide by the rules. Of course, conditional anonymity schemes are unpractical where a high degree of anonymity is desired, but without doubt they can be extremely useful in situations where a fair balance between anonymity and liability is to be enforced. As discussed in the literature (Wolff, 2013), conditional anonymity is usually offered by using the following methods:

- Authentication can be provided by the application itself, i.e., when the user is prompted to provide a credit card number - even if they are not going to be charged any fee - prior to creating a virtual profile.

- Via identity escrow where users need to reveal their real identity to some TTP. The latter has pre-agreed with the service provider that in case of an offensive behavior it will divulge the user's real identity (Mukhamedov and Ryan, 2005). A variation of the aforementioned method is to require multiple TTPs to come to an agreement prior the identification of a user's real identity is possible. For example, in the simplest case, the user entrusts different parts of their encryption key used to encipher their real identity to different TTPs. In this way, no party alone is able to retrieve the user's encryption key and uncover their real identity. Homomorphic cryptographic methods as that of threshold encryption (Fouque et al., 2000) can also provide a workable solution here. Such a scheme ensures that in order one to be able to decrypt a piece of data, a number of participants exceeding a threshold is needed to contribute in the decryption protocol. A characteristic example of this situation in achieving recoverable privacy is given in (Hoepman, 2013).

- Through the employment of the so-called *scoped identities*. The flexibility of having one or more TTPs between the end-user and service is that it becomes possible to reveal only certain, and most relevant

identity elements to the requesting applications. It is then entirely up to the TTP which exact piece of data from the user's profile will reveal depending on the case. For example, it could disclose the sex or the age of an individual but nothing more. However, it is to be noted that although some studies (Lessig, 2006) argue that scoped identities have the power to provide better privacy protections for users over those they enjoy in the real world, the disclose of a critical mass of anonymous identity attributes may, in fact, provide adequate data to allow de-anonymization (Narayanan and Shmatikov, 2008; Ohm, 2009; Tene, 2011). This, once more, clearly designates the need for unlinkability along with anonymity. Note that closely related to the concept of scoped identity is that of *partial identity*, that is, *"an identity of an individual person may comprise many partial identities of which each represents the person in a specific context or role"* (Pfitzmann and Hansen, 2010; Such et al., 2011). Relevant to this discussion is also the solution proposed in the context of EU project ABC4Trust (ABC4Trust, 2014) which is examined in more detail in the next section.

A last major issue regarding identity protection is whether the linking of a single online identity across multiple contexts is advantageous to the end-user or not. This becomes especially important to decentralised systems like OpenID or any sort of federated environment (see next subsection), having that the identity provider is allowed to share reputational information between different contexts (Dolera Tormo et al., 2013). More specifically, it should be clear if abusive user's activity spotted in one context should affect their status in other domains that also use the same identity to authenticate and authorise the user. This may have negative effects on user's privacy, but on the other hand, such a system is able to impose strong accountability mechanisms by means of greatly augmenting the consequences of online misbehavior (Wolff, 2013). Also, this situation is highly probable to render the user more mindful on how they act because any malicious behavior would directly affect the investment they done in the associated identity. Last but not least, schemes like that of *auditable tracing*(Kügler and Vogt, 2002) present an interesting aspect of the problem, but this time from the user's standpoint. Specifically, the aim of such a scheme is to make unauthorised tracing by TTPs detectable in the course of time by the users of the system themselves.

2.3. Identity Management

Today, nearly everyone of us has several online places they need a username and password. Even more, an entity (e.g., a person or organisation) may have zero, one or more identities within a given context. Consider for example the case where a person has two identities in an online store because he is both an employee and a customer at this store. These issues inevitably bring Identity Management (IdM) in the foreground. IdM has to do with the design and administration of users' identity credentials, attributes, and privileges. As a result, IdM consists of two basic parts. The first is responsible of granting users with credentials and IDs upon the initial registration phase, while the latter is in charge of authenticating them and controlling their access to services and resources based on their IDs.

IdM can be carried out in three basic ways; *user-centric*, *federated* or *centralised* (Chadwick, 2009; Cao and Yang, 2010; Maliki and Seigneur, 2013). The first one makes individuals responsible for administrating and controlling any data related to their identities. Identity cards stored on a wallet represents a user-centric type of IdM. For example, the user employs their card to access the university's lab premises. In this respect, the user has the absolute control over how the data on the card are read and used by third parties. Password managers, network anonymisation tools used to curtail exposure of personal information belong to this category as well.

On the other hand, the so-called federated IdM, is a set of standards, agreements and technologies that allow a group of SP to identify user IDs and entitlements stemming from other SPs within a federated domain (in the following the terms domain, realm, ecosystem are used indistinctly). Identity federation is a growing trend among network operators and other communities which aim to offer their Internet services to external end-users. This has the obvious advantage of increasing their business opportunities and consequently their profits. Naturally, in such a domain, trust agreements should be pre-established between SPs prior to identities existing in different domains can be recognised across the federated realm. Lately, however, we came across some research efforts proposing scenarios to shift from the traditional static bilateral agreements to automated dynamic federation, by e.g. based on reputation models (Arias Cabarcos et al., 2013). Some other ongoing proposals towards allowing the dynamic establishment of trust relationships among network domains are that of the so-called Trust Router protocol (along with Temporary Identity Protocol) currently under standardisation by the IETF (Wasserman and Hartman, 2014). This solution enables

the creation of multihop Application Bridging for Federation Beyond the Web (ABFAB) federations without the need of a centralized Public Key Infrastructure (PKI). Similarly, there are research projects like GEANT eduGAIN service (GEANT-project, 2014) and Moonshot (Tschofenig, 2010) (the latter is also under standardisation by IETF ABFAB working group) investigating ways of mitigating the same problem. In any case, this federated IdM eventually results in a single virtual identity domain. So, prior to an end-user is granted access to a service provided by an SP, it is needed to be authenticated by the identity provider (i.e., where this user is registered). Next, authentication and authorisation related information are communicated to the SP, which in turn validates them based on the group of trust relationships agreed between the identity provider and the SP. Nevertheless, the users can still be in charge - at least to some extent - of how their identity credentials are shared and used between the different domains of the federation.

Single Sign-On (SSO) is perhaps the most common use-case in federated IdM, as it enables users to authenticate towards a single site and acquire access to others without supplying additional information or credentials. In this respect, SSO gives answer to two major problems: individuals needing to enter authentication information repeatedly, and individuals having to recall multiple sets of authentication credentials. The Kerberos protocol (see section 3.2) is a characteristic example of such a situation, where the Kerberos Authentication Server (AS) acts as the central point of user authentication and credential provisioning. Microsoft .Net Passport is another example of SSO implementation. Well-known technologies such as OAuth (OAuth, 2013), OpenID (OpenID, 2013), SAML (OASIS, 2005), and WS-Federation are typically used for federating access to Web services. Authentication, Authorization and Accounting (AAA) protocols (e.g., RADIUS, Diameter) along with EAP, to carry out the authentication, are typically the vehicles towards achieving federation of the network access service.

The latest addition to the family of the aforementioned protocols that gathers a lot of attention lately is that of OpenID Connect (OpenID-Foundation, 2014). This comprises a specification that defines how the involved parties can take advantage of the OAuth 2.0 protocol to communicate about identity (i.e., it is built as a simple identity layer on top of the OAuth 2.0 protocol). In this respect, OpenID Connect not only determines the exact way the OpenID 2.0 token and authorization endpoints should interact when authenticating and authorising users under OAuth 2.0, but also details on the way the other - specific to OpenID Connect - endpoints

need to cooperate for exchanging information about the OAuth access token and its holder. This means that OpenID Connect help towards standardising OAuth configurations across different implementations. This in turn promotes interoperability between vendors. The protocol also describes the interplay between the involved parties for registering a client and administering sessions on behalf of the end-user in a dynamic fashion. Overall, by reducing efforts on the developer's side and by being faster, especially for mobile clients due to the use of JavaScript Object Notation (JSON) instead of Extensible Markup Language (XML), it is anticipated that more developers will use OAuth 2.0 to provide secure authentication. A recent report by Gartner (Gartner, 2013) predicted that half of new identities on retail sites will be based on social network identities (Facebook, Google, Twitter etc). This however is feasible only if a common method to easily and securely provide identities from ID providers to SPs to support authentication is gradually established. In this respect, OAuth 2.0 and OpenID Connect seem to be the most promising ones. Bear in mind that an extended reference to the aforementioned Web services federation technologies remains out of the scope of this paper.

Lastly, centralised (or SP-centric) IdM is centrally exercised by others. The simplest way to achieve this is to let a single authority performing as the sole user ID and credentials provider for every other SP. Think for example the case of a PKI being commissioned of issuing certificates to all users within a given domain. When one leaves the organization, their network identity and associated privileges are revoked. Another way to achieve centralised IdM is having service providers to share certain identity related data on a meta (common) level. From a user's point of view, this can be perceived as credential synchronisation across all SPs. Lately, IdM has also started to gain ground as a service in the cloud; this may ideally lead into broaden an organisation's existing IdM capabilities to third-party systems, thus minimising the administration tasks while delivering services to end-users with lowest disruption. In any case, as already pointed out in section 1, identity is eminently contextual; it is meant to be used inside the context it has been issued. So, which type of the aforementioned ID systems is needed each time hinges on the situation at hand.

It is certain that the previous described models are closely tied to the level of anonymity they can offer. In general, when the on demand creation of access credentials is left on the ID provider (as in OpenID, SAML), the ID provider is able not only to track its users but also impersonate them at will.

On the other hand, in systems imposing offline creation of credentials by a TTP (as in X.509 certificates) the user is compelled to uncover a greater set of attributes not necessarily needed by the requesting application. This however renders a user's online movements linkable, say, across the various websites. Obviously, the user-centric model can sustain a high level of anonymity. The rest of the models have to rely on some sort of anonymisation technique (e.g., pseudonymity) to be able to protect user's real identity and mitigate attacks against unlinkability. For example, as pointed out in section 1, traceable anonymous certificates could be used to avoid the use of permanent IDs or even email addresses as unique identifiers in digital certificates issued by a PKI. However, although this is quite feasible on a small scale is rather very difficult to realise on a larger or global one. Also, in Kerberos protocol, where tickets are used for authorising access to services, user's credentials (among others) are visible in every transaction preceding the acquisition of the requested service. So, eavesdropping on which service a given client accesses becomes trivial. This calls for additional measures to be taken in order anonymity to be enforced. This situation is explained in detail in section 3.2.

In this context, an interesting approach of federated IdM has been presented by the EU project ABC4Trust (ABC4Trust, 2014). The motivation of the project contributors remains fundamentally the same: the vast majority of credentials used to authenticate or identify a user is not meant to preserve users' privacy. That is, their identity leaks out despite the requesting application may only ask for much less information. So, their aim is to "address the federation and interchangeability of technologies that support trustworthy, yet privacy-preserving Attribute-based Credentials (privacy-ABC)". Putting it another way, a privacy-ABC allows its holder to reveal only the minimum information needed by the requesting application, thus avoiding the disclosure of full identity information. In short, users obtain privacy-ABCs for their attributes in the same way as any other legacy cryptographic credential, say, a X.509 certificate. This means that a privacy-ABC is signed with the private key of the (trusted) issuer. However, later on, the user is able to self derive unlinkable tokens that reveal only the required attribute information, and more importantly, can be verified using the issuer's public key. In fact, this idea builds on top of other proposals such as that of minimal disclosure tokens, anonymous credentials, self-blindable credentials, group signatures (Belenkiy et al., 2009; Chaum, 2003; Camenisch and Lysyanskaya, 2001). Nevertheless, as already pointed out, the basic aim of this particular

project is the presentation of a language framework enabling a unified deployment of privacy-ABC technologies. Specifically, "the framework offers a set of abstract concepts that make it possible for application developers to set up a Privacy-ABC infrastructure and to author policies without having to deal with the intricacies of the underlying cryptography" (Camenisch et al., 2013).

2.4. Identity Laws and the need for an Identity Metasystem

Relevant to the above discussion is the work in (Cameron, 2006) which introduced seven laws of digital identity for online systems. These are: (a) user control and consent, (b) minimal disclosure of identifying information for constrained uses, (c) disclosure to justifiable parties, (d) directed identities allowing for both public and private identifiers, (f) pluralism of interoperable identity technologies and providers, (g) human integration, and (h) consistent experience across different contexts. The aforementioned laws represent a concrete and valuable framework for understanding and analyzing digital identity models. As commented in (Cavoukian, 2006), when implemented and put into action, these laws can offer to individuals being online great advantages. First off, they enable users to maintain better control over their private information and improve peoples' ability to shrink the amount of identifying data revealed when participating to online interactions. Also, these rules can contribute towards minimizing the correlation between a user's different identities and actions. Lastly, by following these laws, the identification of fraudulent messages and web sites by a user is made easier. Our opinion is that an additional rule should be reckoned. That is, the rule of simplicity and clarity giving the fact that the human factor is at least of equal importance to the technological aspect. It is therefore argued that no digital IdM system would be successful if it is not simple, straightforward, and if possible, transparent to the end-users. This would also have the dual benefit of giving confidence to the users and enabling them to employ it with less errors. For a deeper discussion on these laws the interested reader could refer to several interesting researches in the field of digital privacy like those in (Cavoukian, 2006; Chadwick, 2009; Wolff, 2013; Adjei and Olesen, 2011).

As pointed out in section 1 the adoption of all-in-one digital identity system is very unlikely to happen. This is verified by the variety of - many times contradicting - ad-hoc solutions that comprise the present state of digital identity in the cyberspace. This situation suggests that an identity meta-system is required (Cavoukian, 2006; Cameron, 2006). Such a system

of systems would act as a gateway and thus facilitate the interlinking of all identity systems into a single point of reference. That is, a meta-system could enable us gathering all existing identity systems under the same umbrella as an intermediary; not replace them. Putting it another way, the identity provided by a given system could be used within others irrespective of they are based on the same technologies or not (given of course the existence of such a trusted by all meta-system and after finding ways to overcome the linkability problem). In this way, interoperability between all existing identity systems could be supported, allowing at the same time the build of a unified interface to all of them.

3. Case studies

Taking all the above into account it would be interesting to examine how exactly anonymity is considered and implemented in the context of well-known and widely-deployed Internet protocols such as Session Initiation Protocol (SIP) (Rosenberg et al., 2002) and Kerberos (Neuman et al., 2005; MIT-Kerberos-Consortium, 2014) ones. It is to be noted that although the aforementioned protocols are very different in nature (in terms of what their usage is) they both operate at the application layer and present interesting properties regarding anonymity. Also, both are under constant development and have been adopted in the wired Internet as well as in mobile ecosystems. For instance, since 1988, Kerberos has evolved into a major IETF security standard and surrounded by several other IETF standards and Internet drafts, which are still evolving. As characteristically stated in (MIT-Kerberos-Consortium, 2014) "a conservative estimate of how many are using Kerberos is, probably well over 100 million people, worldwide". On the other hand, one of the main facts that witness in support of the significance and potential of SIP is that 3rd Generation Partnership Project (3GPP) consortium chose it to be the multimedia management protocol of 3G and beyond networks multimedia subsystem (IP Multimedia Subsystem - IMS). This alone is self-evident about the acceptance and future of SIP. Moreover, both of these protocols are not created having primarily in mind the "privacy by design" principle (as explicitly required by the new proposed EU-regulation (E.C., 2012) and others (PbD, 2013). Rather, as expected, they have been built by putting special emphasis on functionality first. Later on, after being put into action, people realised the need for strong anonymity and started to seek novel solutions for meeting this requirement as well. Thus,

we think that the selection of these particular protocols as a dual case study best serves the aim of the current paper.

3.1. The case of SIP

A broad swath of users, including human rights workers, labor organisers, and journalists, is sure to highly appreciate anonymous Voice over IP (VoIP) communications. It is also true that, along with other anonymous services, the provision of anonymous VoIP communications can be proved a great asset for future service providers towards augmenting its pool of users. Here, we focus on SIP as it has been established itself as one of the most prominent protocols supporting multimedia services. SIP is in fact an application layer signaling text-based protocol responsible for the administration of multimedia sessions. Since SIP is an application layer protocol, it can transparently operate over any type of network. However, while SIP is gradually becoming more and more popular, it still suffers from intrinsic privacy issues. The protection of user identities (IDs) is perhaps the most critical of them. That is, virtually everyone is in position to reveal, among others, the communicating parties IDs by simply eavesdropping on the exchanged SIP messages. So, considering the scope of this article, our discussion is confined to proposals aiming to conceal the communicating parties IDs either directly or indirectly. Also, we consider only mechanisms designed to combat unencrypted signaling message attacks (as being the most common and straightforward). As a result, deanonymisation attacks focusing on media flow - as realised over Realtime Transport Protocol (RTP) (Schulzrinne et al., 2003) - are intentionally left out.

As SIP signaling is in plaintext, an eavesdropper is able to very easily read the content of a SIP message and acquire the name and affiliation, IP address or host name, and SIP Uniform Resource Identifier (URI) of the communicating parties. This is possible because the aforementioned information is conveyed by some message header fields, including *From, To, Contact, Call-ID*, used for session establishment. This situation is clearly shown in figure 1, representing the structure of a standard SIP Invite message. Bear in mind that all these pieces of data consist role pseudonyms pertaining to partial identities and are contained in SIP signaling messages and packet headers in plaintext. The Session Description Protocol (SDP) (Handley et al., 2006) body might also reveal the location of a User Agent (UA). For example, a SIP URI is in the form: sip:gkamb@aegean.gr (in figure 1 the username has been replaced by a series of numbers).

Given that a VoIP network can employ either client/server or Peer-to-Peer (P2P) architecture the privacy problem is further complicated by the participation of intermediaries. For instance, SIP proxy servers add their own headers to messages as well. Information contained in these headers, such as <Via> and <Record-Route>, could expose valuable facts about the originator of a message. This situation is characteristically presented in RFC 3323 (Peterson, 2002) which comprises an extension of the basic SIP protocol. Moreover, the work in (Shen and Schulzrinne, 2006) elaborates on optional SIP headers that may leak identity related information, as the real name of a user. Note that for some headers, the caller may be able to hide identity information. However, this is not true in the general case because certain headers are used to route messages inside a session (dialog), so they need to be visible. For ease of discussion, a typical SIP message flow is depicted in figure 2. In the figure also we include the initial user's registration procedure against the *Registrar* as well as other network components needed by SIP UA and proxies. Note that while the figure illustrates the establishment of a call which involves two SIP proxies, any number of proxies can be present depending on the situation at hand.

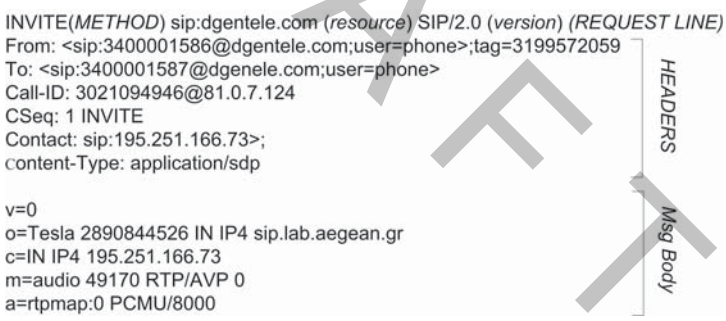

Figure 1: SIP message structure

Identity protection in SIP was somewhat considered RFC 3261 which includes certain mechanisms that can help a user toward protecting their privacy (Rosenberg et al., 2002). These mechanisms can be divided into two major categories; cryptography-based, i.e., Secure/Multipurpose Internet Mail Extensions (S/MIME) (Ramsdell, 2004), SIP over TLS (SIPS) URI/TLS and IPsec, and the non cryptographic solution of "Anonymous" URI (Rosenberg et al., 2002). As already mentioned, a di erent approach is

the extension of the basic SIP protocol which led to the solution presented in RFC 3323. This is in fact a general purpose privacy mechanism which has also been used in RFC 3325 (Jennings et al., 2002) after adaptation. Other major contributions in the same topic in the literature are those given in (Shen and Schulzrinne, 2006; Karopoulos et al., 2011, 2010). All these ID hiding schemes are discussed in the following under the prism of the current work.

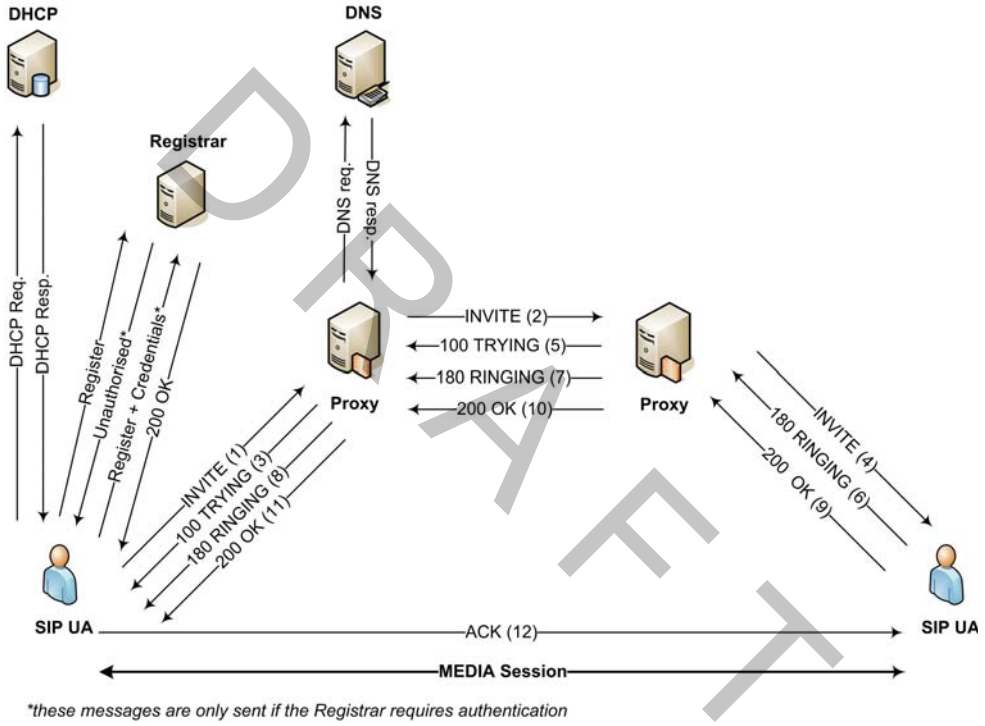

Figure 2: Typical SIP message flow

Before going further, and to better understand this privacy threat in SIP ecosystems, one needs to identify the requirements for enabling user anonymity. This means to find a good mixture of user anonymity and network operability. First off, excluding special cases, any real world anonymity scheme for SIP should support user authentication, which is required among others for accountability and billing. Secondly, the real ID of the user must be available to as less entities as possible. A last demand is that privacy

protection must be assured even through untrusted proxies in an end-to-end fashion. As discussed in section 2.3, to satisfy these needs a pseudonym architecture (or IdM scheme) is required. Such an IdM relies on a trusted identity provider to maintain the association between the real identity and the corresponding pseudonym. Through this mapping, accountability is possible. In the following an up-to-date concise survey of the current anonymity-enabling solutions in SIP is given, followed by a discussion of the findings mainly under the prism of section 2.

3.1.1. Standardisation efforts

The non-cryptographic approach proposed in (Rosenberg et al., 2002) aims at the protection of the caller ID via the use of an anonymous URI in the <From> header (see figure 1). Such a URI can take meaningless values, say, "sip:anonymous@anonymous.invalid". Particularly, this anonymous URI is inserted into the <From> field by the UA itself (user-centric), which means that the SIP proxy cannot access the real URI. The disadvantage of this solution is that it cannot support UA authentication since no user ID is transmitted. As discussed in (Karopoulos et al., 2011) a possible solution to this could be a UA device shared among many end-users. This device will own a specific pair of (username, password) for authentication purposes which will be the same for all users; however such a solution creates other important problems in regards to accountability, repudiation of actions, and billing.

RFC 3323 proposes two types of privacy-enforcing mechanisms; user- and network-supported privacy. The first one is user-centric and is designed having a low-level anonymity in mind. Specifically, any optional personal information contained in SIP messages can be removed by the user. However, this is not adequate as the users' URI and IP addresses are still visible in SIP messages. For dealing with this situation, RFC 3323 describes a centralised, network-oriented privacy facility realised by a TTP acting as a privacy server. This server is in charge of offering transaction pseudonyms by transforming URI contained in SIP messages to randomized sequences. A notable shortcoming of this method is that this entity needs to maintain significant amount of state (session) information, as that of the linking between URIs and the corresponding pseudonym, for the proper routing of messages. As a result, putting aside the risk that this network node is in position to profile the calling records of all users, it also can potentially be a single point of failure in absence of replication. Moreover, this method does not consider any privacy

protections related to the use of standard SIP authentication mechanisms such that of Digest authentication. Nevertheless, a username used in such an authentication method could possibly disclose private information about the end-user. By capitalising on RFC 3323 the work in (Shen and Schulzrinne, 2006), detailed on an enhanced anonymity architecture for SIP, which is still to be implemented and evaluated. Unfortunately, this proposal also suffers from the shortcomings noted previously for RFC 3323.

A more advanced privacy mechanism for SIP is described in informational RFC 5767 (Munakata et al., 2010). This proposal considers signaling as well as media flows and defines an end-users' identity protection framework based on Globally Routable User Agent URI (GRUU) (Rosenberg, 2009) and Traversal Using Relays around NAT (TURN) (Mahy et al., 2010). While TURN provides a temporary IP address for NAT traversal, GRUU acts as an ephemeral globally unique identifier for a specific UA. That is, for a SIP call, a user is able to acquire a temporary URI from a GRUU server and an ephemeral IP address from a TURN one. In this way, role pseudonyms are converted into transaction ones. On the downside, GRUU and TURN are not widely deployed, so the practicality of this proposal - at least for the time being - is limited.

3.1.2. Custom Solutions

The works by (Karopoulos et al., 2011, 2010) (also referred to in the following as PrivaSIP) propose and evaluate two privacy-preserving schemes for SIP based on cryptography. The authors came up with the idea of revealing the user IDs only to the absolutely necessary parties, so as to route SIP messages appropriately and authenticate the caller before service acquisition. In the first scheme, the ID of the caller is protected while in the second both IDs of the caller and callee are protected. Putting it another way, these solutions require SIP service providers to operate as identity providers too. So, when the caller enciphers a header field conveying identity-related information, the service provider is able to recover it by decryption in order to have the message properly forwarded. Specifically, depending on the method, the protection of users' IDs involves the encryption of these IDs and the transmission of their encrypted form instead of cleartext. Through the use of a padding scheme, this encrypted form is a transaction pseudonym and the real ID can be recovered from this pseudonym by entitled entities only.

The authors consider both symmetric and asymmetric key cryptography depending on the case. More specifically, in the first scheme, the caller ID

is protected via either symmetric cryptography, using as a key the digest authentication password shared between the user and their home proxy, or with asymmetric cryptography using the public key of the home proxy. On top of that, when the protection of the callee ID is necessary, the public keys of both the caller's and callee's home domains are used. The most significant advantage of these schemes is that they can assure user ID protection even when SIP messages are transmitted through untrusted SIP domains prior to reaching the home domain of the user or another trusted domain. Moreover, they do not require from the SIP proxy server to maintain state information for the exchanged SIP requests and respective responses. Both these methods support the standard SIP authentication mechanism, namely digest authentication, without revealing the username of the caller to non intended parties. On the other hand, these schemes do not protect domain names and the IP addresses of the communicating parties. A limited usage of PKI is also needed, where digital certificates are issued and managed only for proxies and not for end-users.

3.1.3. Generic and lower level Solutions

As the body part of a SIP message is nothing else than a MIME body, a straightforward solution to protect it is by the use of S/MIME. For safeguarding the privacy of end-users, S/MIME can encapsulate SIP messages into MIME bodies and encrypt them properly. Specifically, the encapsulated message can contain the real ID of the caller, while the outer message contains a <From> header of the form: "sip:anonymous@anonymiser.invalid". When the called party receives the message, it decrypts the body to find the ID of the caller. What must be noted here is that the ID of the callee cannot be anonymised using the same mechanism since the intermediate SIP proxies do not have access to the plain MIME body and thus an anonymous <To> field would make them unable to route the message to the intended recipient. Nevertheless, S/MIME has little practicality in SIP due to some major weaknesses. First off, the receiver of a message must somehow be aware of the identity of the sender beforehand for being able to find the appropriate certificate to decrypt the message body. Also, the receiver knows the ID of the sender, while the receiver's ID is not protected from third parties. S/MIME cannot support authentication since the ID of the caller is not visible to any of the intermediate SIP proxies. Finally, this solution requires the full deployment of a PKI to manage certificates for the end-users.

To ensure their privacy protection, end-users may request that their mes-

sages along the whole path (hop-by-hop) to its destination are transported inside a TLS tunnel. Although TLS can be employed in each hop, it is not possible to instruct or even be informed that it will be used in every intermediate link; so, end-to-end protection is not assured. Moreover, the employment of TLS normally implies the use of TCP as a transport means, while the legacy transport protocol for SIP is UDP. Note that Datagram Transport Layer Security (DTLS) (Rescorla and Modadugu, 2012), which is the equivalent of TLS using UDP as transport mechanism, is not included in (Rosenberg et al., 2002). SIPS also requires the operation of a full PKI to administer digital certificates for end-users and intermediate SIP proxies.

Going a layer deeper in the Internet stack, one can argue that IPsec can be a strong candidate for protecting SIP signaling in a fully transparent for the end-user manner. Indeed, as stated in (Rosenberg et al., 2002), IPsec is a more suitable in cases where the communicating hosts have already established a trust relationship with one another as opposed to SIPS URI scheme. Unfortunately, what stands true for end-to-end protection in SIPS also applies here; it is not guaranteed. However, IP protection can be imposed if proxy-to-proxy communications are realized by the use of Encapsulating Security Payload (ESP) in tunnel mode.

3.1.4. Discussion

Taking all the above into account we can argue that SIP consists a quite interesting case as to preserving user anonymity. It can be observed that already from RFC 3261 (Rosenberg et al., 2002) some identity hiding methods have been proposed. However, the three of them, namely S/MIME, SIPS, and IPsec, are generic to any application, not specific to SIP. Over the years, two more RFCs arose along with other custom-tailored solutions proposed by various researchers. An interesting observation is that none of the above mentioned schemes requires changes to the basic SIP protocol. Note that an exhaustive comparison of all these privacy-preserving methods remains out of the scope of this paper and the interested reader can refer to (Karopoulos et al., 2011, 2010; Zhang and Fischer-Hubner, 2013). Yet, a discussion of the findings under the umbrella of this work is needed.

It is apparent that protecting the end-user's ID is not a trivial task. This actually verifies what was emphasised in sections 1, 2; that is, digital identity is primarily related to context. Therefore, in the case of SIP, the following interlinked observations emerge:

- *Level of anonymity*: One needs to make some logical assumptions regarding the parties that would be able to access the ID of a given user. For example, should the ID of some user be available merely to himself and his home domain or only the owner of the ID will have access to it? This is crucial in order to decide which user's sensitive information contained in SIP headers needs to be protected and in which particular case.

- *Anonymity vs. pseudonymity*: This issue has to do with what is stated in section 2.1. A person receiving a call from a UA using a pseudonym can always return the call using the same pseudonym. This is not feasible with totally anonymous schemes. Also, pseudonymity means that all protected IDs are recoverable by the corresponding SIP entities, which are the home servers of both parties. This means that data retention policies do not need to change; service providers can log connection information and recover a user ID upon request.

- *Accountability*: Further to the previous point, if the anonymity scheme does not support the standard SIP registration process then accountability (and billing) cannot be enforced.

- *Cryptography*: Some schemes rely on cryptography to keep personal information private while others employ other means. Those that do not use cryptography will probably be faster and have less administrative requirements in terms of key management.

- *Deployment cost*: This criterion has to do with the easiness of deployment of a scheme. For instance, a scheme that requires the full deployment of PKI, as that of S/MIME, presents a high cost.

- *Depth of protection*: An answer on whether a scheme is capable of protecting user's ID private information leaking from other layers (apart from the application one) is required.

- *User-centric vs. Centralised IdM*: Two of the methods can be characterised as user-centric; Anonymous URI, and partly the one given in RFC 3323. SIPS and IPsec are based on the construction of secure tunnels and thus IdM with reference to SIP is not applicable to them. All the others offer centralised IdM.

- *Overhead*: This refers to the overhead imposed by the solution at hand. Leaving aside the increased resource consumption caused to servers, in SIP, a critical parameter is that of the user's service time (latency). This parameter is only discussed in solutions (Karopoulos et al., 2011, 2010) and, as expected, was found to be closely related to the selected cryptographic scheme. The use of a symmetric algorithm like AES, resulted in insignificant delays. In contrary, a SIP request preparation delay may be increased to over 45 milliseconds (ms) per message operation in case of asymmetric algorithms (Karopoulos et al., 2010). On the other hand, the mean server response delay for a SIP server having a queue size of 1000 calls may be increased up to a maximum of 800 ms.

Taking the above into account, we can provide a short but comprehensive comparison between the solutions described in the three previous subsections. Sometimes the caller wishes not to reveal their ID to the callee. This ID hiding option is offered by the PrivaSIP methods (Karopoulos et al., 2011, 2010), "Anonymous URI", RFC 3323, and RFC 5767. Nevertheless, only the PrivaSIP ones are able to afford this feature while protecting the Digest username during the authentication process at the same time (see figure 2. Also, the same methods are in position to keep their privacy protecting features active while operating through untrusted domains. S/MIME can also protect the user ID, still it is unable to protect their username during authentication. Moreover, it cannot offer caller's ID hiding from the callee. The protection of the home domain name of the caller can be only achieved by the use of "Anonymous URI". However, as explained in section 3.1.1, this method has little practicality since it cannot support authentication. Regarding the IP addresses of the communicating parties it is evident that no method except SIPS, RFC 5767 and IPsec ones can effectively protect them from eavesdroppers. RFC 5767 and IPsec (under certain mode and algorithm of operation) can also provide privacy protection down to IP layer. Still, one has to carefully consider the special network architecture required by the RFC 5767 solution and the deployment cost imposed by the IPsec one.

Another observation is that no scheme is able to protect the callee ID only. This may be useful for example in cases a user calls a certain hotline. As mentioned in section 1 this scenario could be of value for service providers as the available information to an observer would be "user U is calling a

service from domain D2". Persistent traffic analysis is also not considered by any of the above schemes. This, however, consists an important threat as in any VoIP ecosystem the probability of a call between a couple of users does not occur at a uniform rate, given that each user usually has a different set of contacts. So, de-anonymisation attacks via the exploitation of long-term statistic information are made possible (Zhang and Fischer-Hubner, 2013; Danezis, 2003).

A last point of discussion, which is also brought up later on in section 3.2.4, is the utilisation of some anonymous communication system, like the well-known Tor, to maximize the level of obfuscation achieved regarding SIP messages. Note that such anonymisation systems are self-reliant, i.e., usually its operation does not depend on the protocols of upper layers, hence they can be seamlessly combined with them. Taking Tor as an example, the problem with SIP is that currently Tor only supports TCP for its transport layer. So, although (Rosenberg et al., 2002) requires all SIP entities to mandatory implement both UDP and TCP, many real-world VoIP applications rely solely on UDP for latency reasons. So, at least for the time being, this is a serious impediment for VoIP users to enjoy strong anonymity to real-time voice communication. Tunneling of the UDP traffic through Tor does not really solve this issue because the traffic would be encapsulated in TCP. The latency induced by Tor is also increased as the system relays and mixes its traffic via multiple nodes. Despite that, recently, a first effort to realise VoIP over Tor was materialised in an opensource product called Torfone (TorFone, 2013). However, Torfone is not based on SIP but on an (obsolete) version of zfone (ZRTP) (http://zfoneproject.com/). Its implementors do recognise this latency problem by stating that *"The payment for anonymity is voice latency up to 2-4 seconds"*. This observation is roughly verified by some early and still ongoing experimental results of ours showing an additional mean latency of about 700 ms when routing SIP traffic over Tor. This time penalty is associated only to SIP signaling and it is perceived starting from the moment the caller's UA sends out an invite until an OK message is received by her. In any case, this is a quite interesting research issue and it is sure to gain momentum as Tor network performance increases over time, and some day it will eventually support UDP as well. For the interested reader, a detailed analysis of similar to Tor solutions can be found in (Edman and Yener, 2009; Ren and Wu, 2010; Ruiz-Martinez, 2012).

3.2. The case of Kerberos

As already pointed out, the Kerberos protocol (Neuman et al., 2005) is quite different in nature as compared to SIP. In fact, Kerberos can be seen as a service used for other services to authenticate users. Nowadays, Kerberos is one of the most well-established three-party authentication and key management protocols over open and insecure networks (MIT-Kerberos-Consortium, 2014). The protocol offers a SSO platform through the use of tickets, i.e., a piece of enciphered and integrity protected information that enables a user to be authenticated without re-entering their password. By capitalising on the extensive adoption of Kerberos by modern application services, Kerberos is also starting to gather considerable attention as a solution to provide federated access to any kind of application service through AAA infrastructure (Perez-Mendez et al., 2013).

The standard Kerberos protocol lacks of a mechanism to preserve user privacy. More precisely, in a similar way to SIP, Kerberos identifies the different participant entities via identifiers, which are in the form of "principal@realm". For example, gkamb@AEGEAN.GR and printer/server.aegean.gr@AEGEAN.GR are valid IDs of a user and a service respectively within the AEGEAN.GR realm. Unfortunately, these principal IDs associated to both clients and services are communicated in cleartext. More specifically, the service identifier for which the ticket has been issued is conveyed in cleartext. Even more, the two messages of the AS exchange (Neuman et al., 2005) contain the client's ID which is being authenticated by the AS module of the Key Distribution Center (KDC). The identity of the service the client is willing to access is also visible to any eavesdropper when monitoring the Ticket Granting Server (TGS) exchange (Neuman et al., 2005). Undeniably, this situation clearly violates the principle of user anonymity as an observer can straightforwardly learn the client's real ID and discover which services are being accessed by them. Figure 3 depicts a typical Kerberos message flow concerning both single and cross-realm operation.

The basic service access model defined in Kerberos also contributes to privacy violations. This is due to the Kerberos atomic operation where the client first performs a message exchange with the KDC to acquire a ticket that is used in a subsequent exchange to access a service. In fact, this situation is present in the AS exchange (to provide the client a Ticket Granting Ticket (TGT) to be used in the TGS exchange) as well as in the TGS exchange (needed for the client to obtain a Service Ticket (ST) that is delivered

to the service via the AP exchange). So, an eavesdropper is able to easily correlate the different messages sent or received by a client to access a service. Even worse, a listener is in position to collect information about behavioral patterns of service access of given users in the network. This is true because typically the acquisition of an ST by a client to access a service is performed via the use of the same TGT used previously for obtaining other services. This simply means that service access unlinkability is not preserved, as by tracing the use of a given TGT, a malicious actor can figure out that the same - even anonymous - client is accessing these services (Tene, 2011; King and Jessen, 2010).

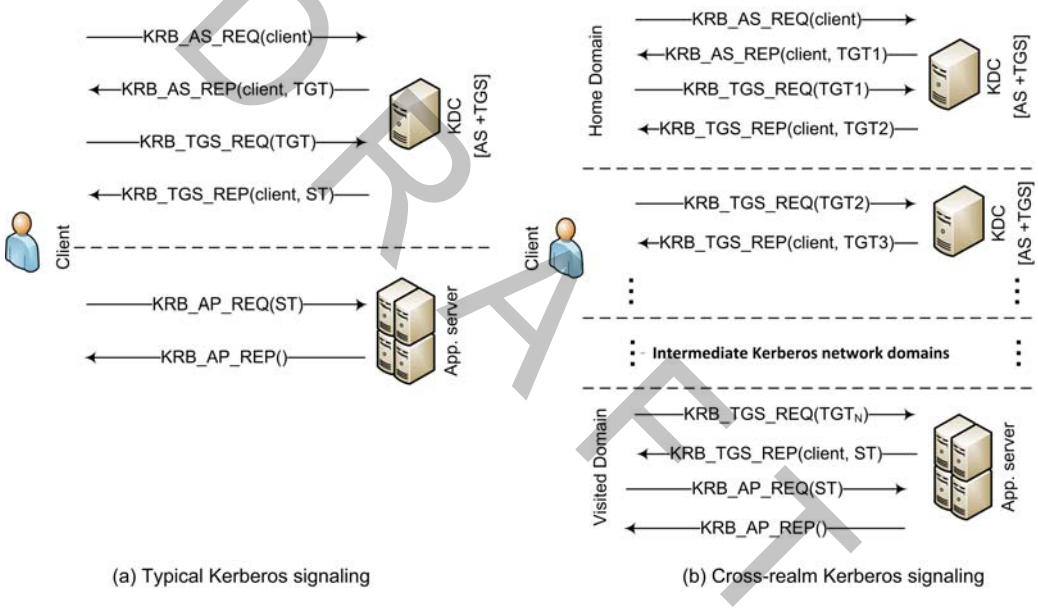

Figure 3: Kerberos protocol message flow

Taking all the above into account, to achieve user anonymity in Kerberos one must (a) guarantee that the real identifier (e.g., the username) of a given user involved in Kerberos transactions remain hidden, and (b) prevent malicious entities from being able to cross-relate the different messages sent and/or received by a specific user, thus providing message exchange unlinkability. Naturally, this anonymity facility is better to include multi-domain (e.g., federated) Kerberos environments as well. In the next subsections, a concise survey of the current privacy-enabling solutions in Kerberos is offered,

followed by a discussion of the findings.

3.2.1. Standardisation efforts

In an attempt to offer user anonymity, the work in (Medvinsky et al., 1998; Zhu et al., 2011) enhances Kerberos protocol by introducing the anonymous ticket concept. Instead of being assigned to a specific user registered in a realm, an anonymous ticket is associated to the anonymous user (anon@anon). Therefore, the true client identity is not revealed neither to the service nor to eavesdroppers. This, of course, pertains to a purely anonymous scheme rather a pseudonymous one. However, for the anonymous TGT acquisition, this solution requires the utilisation of anonymous Public Key Cryptography for Initial Authentication in Kerberos (PKINIT) (Zhu and Tung, 2006). Specifically, for the KDC to securely deliver to the client the session key associated to the anonymous ticket, a secure channel needs to be established by making use of certificate-based KDC authentication. This leads to a PKI-depended solution which is not usually available, especially in multi-domain environments. Moreover, the user remains completely anonymous even to KDC and services which are considered trusted. This situation bears accountability problems as the organization controlling the foreign domain typically needs for example to charge the visited user for the services they obtained.

The Generalized Framework for Kerberos Pre-authentication (Hartman and Zhu, 2011) comprises another solution towards addressing the client privacy issue in Kerberos. This framework is concerned with the protection of client's identity in the messages transmitted from the client to KDC, in such a way that the use of anonymous PKINIT is not required. Simply put, by combining the anonymous ticket concept with the security extensions defined in (Hartman and Zhu, 2011), clients are able to obtain anonymous tickets. On the downside, this procedure mandates the acquisition of a special ticket called armor TGT (contains a symmetric key known to both the client and KDC to protect Kerberos exchanges), which in turn, presents certain deficiencies. Note that an armor TGT must obtained before a client starts to utilise the pre-authentication extensions with a given KDC. Specifically, three solutions are proposed. First, using its real identity, a client is able to exercise a standard AS exchange to request an armor TGT. This however allows listeners to easily correlate the client's ID with the acquired armor TGT, meaning that when the client employs the armor TGT towards requesting an anonymous ticket, an eavesdropper

can derive their real identity associated to the anonymous ticket. Second, the user can obtain an armor TGT via the use of anonymous PKINIT. This of course requires the KDC to own a valid certificate, which in turn requires PKI. In case a PKI infrastructure is not present, a final method is to acquire the armor TGT using anonymous PKINIT without KDC authentication. Nevertheless, as stressed out by the authors, this option is prone to man-in-the-middle attacks.

3.2.2. Custom Solutions

The work in (Pereñiguez-Garcia et al., 2011) proposes a privacy framework for Kerberos, coined as "PrivaKERB". This framework does not require the existence of PKI or other infrastructure external to Kerberos. A prominent feature of PrivaKERB is that along with user anonymity it offers service access unlinkability. The latter refers to the granting of tickets problem pointed out in section 3.2, that enables eavesdroppers to trace the different services accessed by a specific user. It is to be noted that the identity-enhancing functionalities by this work remain in total harmony with user identification required by processes such as accounting and charging. Particularly, to deliver user anonymity, the authors make use of temporary client pseudonyms (transaction pseudonyms) only valid for a specific period of time. The KDC is in control of pseudonym generation and a new pseudonym is mandatorily delivered to the client each time a fresh home TGT is issued. Moreover, service access unlinkability is imposed via the use of extended anonymous tickets which include the client's pseudonym in such a way that is only accessible by trusted parties (i.e., KDCs, services). To obstruct the TGT-based linkability that takes place when a client reuses the same TGT several times to solicit access to multiple services, the authors propose a new kind of single-use TGT called "self-renewed TGT". Summarizing, this solution accomplishes a fair level of unlinkability that prevents eavesdroppers from linking the different service accesses performed by a specific anonymous user. This way, attackers cannot deduce whether separate service accesses belong to the same or different anonymous clients.

On the negative side, PrivaKERB contributes little in protecting users from observers attempting to cross-link the sequence of different messages communicated by a specific - even anonymous - user to acquire a service. Indeed, the series of messages starting with *TGT acquisition from the KDC* and followed by *ST obtainment from the KDC* and *ST handing over to the service* can be associated to the same anonymous user, and thus, exploited

by attackers to reveal the different Kerberos transactions in which a client participated. This however leads to a privacy breach in multi-domain scenarios as potential eavesdroppers have the chance to easily find out the service visited by a client in a remote Kerberos domain. If these pieces of data are systematically collected, it can be used toward disclosing important information such as the most preferred services for roaming users in a realm and/or the origin realm of clients visiting a specific Kerberos domain.

Motivated by the aforementioned insufficiency, the same authors contributed a full-fledged anonymity framework, called KAMU (Pereñiguez-Garcia et al., 2013), able to achieve a full obfuscation of the protocol's messages from an eavesdropper point of view. To fix the foregoing linkability problem, the authors came up with the specification of a mechanism able to obscure the KDC distributed tickets, so as to hinder observers from tracking the different tickets acquired by a client. This mechanism camouflages by means of encryption the ticket (TGT or ST) sent by the KDC to the client. By doing so, eavesdroppers cannot observe the ticket and only the client can recover it. The implementation of this solution is based on both normal Kerberos tickets as well as a new type of ticket, called "fake ticket". The solution requires normal tickets to be transmitted in a way that remain confidentiality and integrity protected, rather than having certain parts of them being visible (recall that in standard Kerberos the service's ID for which, say, an ST ticket has been granted is transmitted in cleartext). In this way, attackers are blocked from accessing or modifying a ticket. Also, according to this solution, a normal ticket is placed in the padata field (Neuman et al., 2005) of the message. This is an extensible field defined by Kerberos with the aim to develop new functionalities or convey additional data.

On the other hand, a fake ticket is an entirely new type of ticket having all of its fields belonging to the protected part (named EncTicketPart (Neuman et al., 2005)) contain meaningless (null or randomly initialized) information. This however does not apply to the flags field; this is done to enable all entities to recognise a fake ticket from a standard one by the presence of the fake flag. A fake ticket is intended to replace the standard one, and thus, it is placed in the normal ticket field (Neuman et al., 2005) of every reply message issued by the AS or TGS. As a consequence, no one except the authorised entities are capable of accessing the real TGT or ST being communicated to the client.

3.2.3. Lower level Solutions

As in the case of SIP, one apparent solution to deliver anonymity in Kerberos could be the use of TLS tunnels (Josefsson, 2011). This way, eavesdroppers would be blocked from snooping on sensitive information such as the client and service identifier. It is obvious thought that this solution presents the limitation of the hop-by-hop requirement; that is, the creation of a TLS tunnel between every pair of communicating entities is needed. This is however particularly defective in multi-domain scenarios as the clients would need to establish a TLS session with every intermediary KDC in the path from the home to visited domain. Still, no one can guarantee (or even acknowledge) that every network hop does afford (or establish) a TLS tunnel. A multi-domain PKI (Shimaoka et al., 2008) has also to be in place, since a pre-established trust relationship between the client and intermediary realms is not usually the case. These requirements are sure to significantly increase the deployment cost of the solution. Naturally, as already pointed out in section 3.1.3, the same shortcomings are to be taken for granted for lower layer tunnels, as that of IPsec.

3.2.4. Discussion

From the above discussion it becomes apparent that the tackling of the anonymity and unlinkabilty problem in Kerberos presents many similarities to that of SIP. It can be said that several critical privacy insufficiencies have been identified and a considerable mass of works are devoted in solving the problem. As in the case of SIP, we can perceive some standardisation efforts along with custom and generic solutions. Nevertheless, once more, it can be argued that generic solutions are just passing the privacy problem to a lower layer's protocol (e.g., TLS, IPsec). As already highlighted, while this solution works for virtually every superjacent protocol in the stack, it presents certain shortcomings mainly due to the need of external infrastructures and the hop-by-hop impediment. So, while none of the aforementioned solutions requires changes to the core Kerberos protocol, some of them are fully or partially based on PKI, which unfortunately - at least until now - is not the case for the majority of network realms.

Using the key points already identified in section 3.1.4 and the discussion given in sections 1 and 2 we can notice the following qualities:

- *Level of anonymity*: As pointed out in section 3.2, all principal IDs associated to both clients and services are visible to an observer when

in transit. So, any solution should guarantee that no one except the user herself and the service has access to their real identifiers. This is however a basic (or first) level of anonymity because due to the atomic operation of Kerberos explained in section 3.2 and illustrated in figure 3, a malicious actor is able to cross-relate information contained in the protocol's message flow, and thus identify a client even in case they remain anonymous. Therefore, as discussed in section 2.1, to obtain a stronger level of anonymity in Kerberos one requires to also impose message unlinkability. Furthermore, any given solution needs to consider single- as well as multi-realm network deployments in a way that no sensitive information is leaked out even in cases where some or all of the intermediate realms in the path collude. Once again, as the provision of anonymity and the protection of digital ID is primarily related to context, another important property for any solution here is to be able to support all levels of anonymity in an opt-in basis. Indeed, this property seems to be satisfied by both PrivaKerb and KAMU. Lastly, lower level solutions are offering both anonymity and unlinkability, but unfortunately they provide a lesser degree of flexibility and present certain shortcomings that need to be laboriously evaluated prior to deployment.

- *Anonymity vs. pseudonymity*: This issue has mainly to do with what has been discussed in section 2.1. So, while a totally anonymous solution have been proposed for Kerberos (Medvinsky et al., 1998; Zhu et al., 2011) it seems that it is conflicting with accountability. This is because the client does not reveal their ID even to KDC and the service which are considered a priori trusted. Also, as discussed in section 3.2.1, this proposal requires PKINIT. The Generalized Framework for Kerberos Pre-authentication tries to solve this latter issue but creates another deficiency pertaining to unlinkability. On the other hand, the two custom solutions namely PrivaKERB and KAMU are based on preudonymity. However, in relation to section 2.1, these solutions make use of transaction pseudonyms at the client side which is not really "one-time identity" but valid for a specific period of time. In addition, KAMU exercises an interesting camouflaging ticket scheme along with encryption to work-around the problem and achieve full obfuscation of the protocol's message flow.

- *Accountability*: Elaborating on the previous point, and under the umbrella of section 2.2, totally anonymous solutions cannot support accountability which is normally required by the service provider. In special cases, where for example a service is given to clients for free, full anonymity may be desirable (and thus the existence of such a scheme would become very handy). Nevertheless, a mechanism to elevate back and forth to an accountable state is usually needed in this case, which in turn adds complexity to the system, and thus such a decision is normally densely interwoven with the particular case at hand.

- *Cryptography*: As discussed in 3.2.3 the lower level solutions which are in charge of constructing a secure tunnel for channelling all protocol's sensitive information through it are ordinarily impose heavier cryptography compared to those working entirely at the application layer. Namely, the higher the layer of protection the greater the level of customisation. In this respect, custom solutions such as KAMU employ tailor-made strategies to protect only the information that matters. This is verified by the results reported in the context of these works and briefly outlined further down. Lastly, bear in mind that a main solicitude of the solutions discussed in section 3.2.2 was critical tasks, like that of pseudonym generation, to be consigned to KDC care in an effort to discharge the client from frequent operations that add overhead.

- *Deployment cost*: As in the case of SIP, several solutions proposed for Kerberos impose the use of some sort of PKI. This however comes at a high cost and makes deployment far from being simple. Flexibility and compatibility with current implementations is also key issues here towards building a truly workable solution.

- *Depth of protection*: As with SIP, Kerberos works at the application layer, so the protection of private information about a user disclosed by other layers is needed. From the foregoing discussion it becomes glaring that apart from the SSL and IPsec solutions, all the others cope with privacy at Kerberos level only. This situation results in the same worriment spotted for SIP in section 3.1.4 (also briefly sketched in section 1). So, the question here is what happens with privacy-sensitive information belonging to the TCP/IP layer over which Kerberos is conveyed? IP address, ports, domain name, packet contexts, sizes and timing, and round-trip times are only certain pieces of information that can be

used towards identifying the communicating parties. Naturally, this is a cross-layer privacy problem that calls for solutions like that of Tor. This is recognised by the authors in (Pereñiguez-Garcia et al., 2013) which point out several possible solutions to integrate with KAMU. They particularly focus on Tor and explain that the formation of an alliance between KAMU and Tor results in a robust cross-layer privacy-preserving communication system able to effectively deal with a considerable number of privacy attacks.

- *User-centric vs. Centralised IdM*: Under the scope of section 2.3 all the anonymity solutions discussed in the present section offer centralised IdM. Of course, as in SIP, TLS and IPsec driven solutions do not provide any kind of IdM Kerberos intrinsic solution.

- *Overhead*: To our knowledge, experimental results about the penalty in terms of service times are available only for PrivaKERB and KAMU. Specifically, a ccording t o t he a uthors o f t hese w orks, t he fi rst on e is found to augment the AS and TGS exchange time by 0.35 ms and 1.4 ms respectively. As expected, KAMU produces higher overheads due to the extra time required to distribute the reinforced in terms of privacy ticket. This however is translated to an negligible latency of about 0.89 and 2 ms for AS and TGS exchange correspondingly. Also, compared to the message processing time for standard Kerberos, KAMU produces insignificant o verheads. F or i nstance, i n T GS e xchange which represents the worst-case, the authors recorded an increment of 1.1 and 0.3 ms for the client and the KDC respectively. For further details on these metrics and experimental results the reader can refer to (Pereñiguez-Garcia et al., 2011, 2013).

In summary, the KAMU solution seems to be the most complete in regards to its privacy features. It not only allows the client to remain anonymous and untraceable from eavesdroppers, but also does not hinder the identification of clients when needed, e.g., for accounting and charging processes. Moreover, its privacy features are preserved in both single- and multi-domain scenarios without the need of PKI, as it simply relies on existing Kerberos extensibility mechanisms. On the other hand, works like (Zhu et al., 2011; Medvinsky et al., 1998) attempt to render the client fully anonymous and thus fail to support important accounting operations performed by trusted entities.

4. Conclusions

The need for anonymity is inevitably present in almost any protocol, application or service used in wired or wireless networks. Undoubtedly, the need of being innominate is an issue of great importance as it comprises the basis to protect fundamental human rights, such as the free expression of ideas and opinions, and allow people to perform their online activities in comfort and privacy. In this context, the goal of this paper is twofold. First off, it sheds light on the various issues revolving around anonymity and argues that it is a versatile concept that includes and affects several others, such as that of accountability, linkability, identity management, and so forth. Secondly, it conducts a short but comprehensive survey on the anonymity-preserving solutions proposed so far in the literature regarding SIP and Kerberos protocols. This serves as a dual case study for investigating the ways user's anonymity, and more general privacy, is confronted and dealt in the context of major, well-established protocols used at large in the cyberspace. In this respect, the survey part of the work at hand differs from the great mass of earlier ones which particularly focus on Web anonymity or anonymisation tools.

Also, bear in mind that the choice to include two application layer protocols (and not another from a lower layer) is not taken without due consideration. This is because providing anonymity and privacy in general at the application layer is usually harder to achieve and therefore more interesting. That is, any proposed solution needs to be tailored to the application, support accountability, and retain compatibility with current implementations. Moreover, we have in mind the case studies to be somehow comparable to each other. This would problematic if we choose protocols lying in different layers of the Internet stack. It is therefore really interesting to observe that although the two aforementioned protocols have totally different usage, they was found to utilise quite similar methods to address user's privacy.

Moreover, this study has confirmed that every anonymity-preserving solution considers either directly or indirectly, and at least to some degree, aspects like accountability as those have been identified in the first part of the current work. It has been also exhibited that the research on this topic is active and constantly growing as anonymity and privacy in general for most protocols and services have not been treated in a "by-design" fashion. In this context, it seems that the most noteworthy issues for future designs to deal with is that of o
ering cross-layer privacy-preserving systems, the compatibility with

the base protocols, the support of anonymity across different, but somehow federated network realms, and the smooth integration of anonymity with vital underlying network operations.

References

ABC4Trust, Jan. 2014. Eu project - attribute-based credentials for trust. URL https://abc4trust.eu

Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., Levkowetz, H., June 2004. Extensible authentication protocol (eap). IETF RFC 3748.

Adjei, J., Olesen, H., 2011. Keeping identiy private. IEEE Vehicular Technology Magazine 6 (3), 70–79.

Arias Cabarcos, P., Almenarez, F., Gomez Marmol, F., Andres, M., 2013. To federate or not to federate: A reputation-based mechanism to dynamize cooperation in identity management. Wireless Personal Communications, 1–18.

Belenkiy, M., Camenisch, J., Chase, M., Kohlweiss, M., Lysyanskaya, A., Shacham, H., 2009. Randomizable proofs and delegatable anonymous credentials. In: CRYPTO. pp. 108–125.

Burkell, J., 2006. Anonymity in behavioural research: Not being unnamed, but being unknown. University of Ottawa Law & Technology Journal 3 (1), 189–203.

Camenisch, J., Dubovitskaya, M., Lehmann, A., Neven, G., Paquin, C., Preiss, F.-S., 2013. Concepts and languages for privacy-preserving attribute-based authentication. In: IDMAN. pp. 34–52.

Camenisch, J., Lysyanskaya, A., 2001. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In: EUROCRYPT. pp. 93–118.

Cameron, K., Jan. 2006. The laws of identity. URL http://www.identityblog.com/?p=354

Cao, Y., Yang, L., 2010. A survey of identity management technology. In: IEEE International Conference on Information Theory and Information Security (ICITIS). pp. 287–293.

Cavoukian, A., Oct. 2006. 7 laws of identity - the case for privacy-embedded laws of identity in the digital age. White paper, Information & Privacy Commissioner, Ontario, Canada.
URL http://www.ipc.on.ca/images/Resources/up-7laws_whitepaper.pdf

Chadwick, D. W., 2009. Federated identity management. In: Aldini, A., Barthe, G., Gorrieri, R. (Eds.), Foundations of Security Analysis and Design V. Vol. 5705 of LNCS. Springer Berlin Heidelberg, pp. 96–120.

Chaum, D., 2003. Untraceable electronic mail, return addresses and digital pseudonyms. In: Secure Electronic Voting. pp. 211–219.

Danezis, G., 2003. Statistical disclosure attacks. In: of the IFIP TC11 18th International Conference on Information Security (SEC '03). Kluwer, pp. 421–426.

Davenport, D., 2002. Anonymity on the internet: why the price may be too high. Commun. ACM 45 (4), 33–35.

Dolera Tormo, G., Gomez Marmol, F., Martinez Perez, G., 2013. Towards the integration of reputation management in openid. Computer Standards & Interfaces In Press, Accepted Manuscript, –.

E.C., 2012. Proposal for a regulation of the european parliament and of the council on the protection of individuals with regard to the processing of personal data and on the free movement of such data (general data protection regulation). COM(2012) 11 final.
URL http://ec.europa.eu/justice/data-protection/document/review2012/com_2012_11_en.pdf

Edman, M., Yener, B., 2009. On anonymity in an electronic society: A survey of anonymous communication systems. ACM Comput. Surv. 42 (1).

E.U., Oct. 1995. European parliament - protection of individuals with regard to the processing of personal data and on the free movement of such data.
URL http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:EN:HTML

Fouque, P.-A., Poupard, G., Stern, J., 2000. Sharing decryption in the context of voting or lotteries. In: Financial Cryptography. pp. 90–104.

Gartner, Feb. 2013. Half of new retail customer identities will be based on social network identities by 2015.
URL http://www.gartner.com/newsroom/id/2326015

GEANT-project, 2014. edugain service.
URL http://www.edugain.org

Handley, M., Jacobson, V., Perkins, C., July 2006. Sdp: Session description protocol. IETF RFC 4566.

Hansen, M., Tschofenig, H., Smith, R., Oct. 2011. Privacy terminology and concepts.
URL http://tools.ietf.org/html/draft-hansen-privacy-terminology-03

Hartman, S., Zhu, L., April 2011. A generalized framework for kerberos pre-authentication. IETF RFC 6113.

Hoepman, J.-H., May 2013. Revocable privacy.
URL http://www.cs.ru.nl/~jhh/revocable-privacy/index.html

ITU-T, Jan. 2009. Ngn identity management framework. Recommendation Y.2720.

Jennings, C., Peterson, J., Watson, M., Nov. 2002. Private extensions to the session initiation protocol (sip) for asserted identity within trusted networks. IETF RFC 3325.

Johnson, C. Y., Sept. 2009. Project gaydar.
URL http://www.boston.com/bostonglobe/ideas/articles/2009/09/20/project_gaydar_an_mit_experiment_raises_new_questions_about_online_privacy/

Josefsson, S., May 2011. Using kerberos version 5 over the transport layer security (tls) protocol. IETF RFC 6251.

Karopoulos, G., Kambourakis, G., Gritzalis, S., 2011. Privasip: Ad-hoc identity privacy in sip. Computer Standards & Interfaces 33 (3), 301–314.

Karopoulos, G., Kambourakis, G., Gritzalis, S., Konstantinou, E., 2010. A framework for identity privacy in sip. J. Network and Computer Applications 33 (1), 16–28.

Kim, M., 2010. The right to anonymous association in cyberspace: Us legal protection for anonymity in name, in face, and in action. SCRIPTed 51 7 (1), 51–70.

King, N. J., Jessen, P. W., 2010. Profiling the mobile customer–privacy concerns when behavioural advertisers target mobile phones–part i. Computer Law & Security Review 26 (5), 455–478.
URL http://www.sciencedirect.com/science/article/pii/S0267364910001044

Kügler, D., Vogt, H., 2002. Offline payments with auditable tracing. In: Financial Cryptography. pp. 269–281.

Lessig, L., 2006. Codev2. Basic Books.
URL http://www.codev2.cc/download+remix/Lessig-Codev2.pdf

Mahy, R., Matthews, P., Rosenberg, J., April 2010. Traversal using relays around nat (turn): Relay extensions to session traversal utilities for nat (stun). IETF RFC 5766.

Maliki, T. E., Seigneur, J.-M., 2013. Online identity and user management services - chapter 25. In: Computer and Information Security Handbook (Second Edition), second edition Edition. Morgan Kaufmann, Boston, pp. 459–484.

Medvinsky, A., Cargille, J., Hur, M., March 1998. Anonymous credentials in kerberos. IETF Internet Draft.
URL http://tools.ietf.org/html/draft-ietf-cat-kerberos-anoncred-00

MIT-Kerberos-Consortium, Jan. 2014. Mit kerberos & internet trust (mit-kit) consortium. MIT Kerberos & Internet trust (MIT-KIT) Consortium.
URL http://www.kerberos.org/

Mukhamedov, A., Ryan, M. D., 2005. On anonymity with identity escrow. In: Formal Aspects in Security and Trust. pp. 235–243.

Munakata, M., Schubert, S., Ohba, T., April 2010. User-agent-driven privacy mechanism for sip. IETF RFC 5767.

Narayanan, A., Shmatikov, V., 2008. Robust de-anonymization of large sparse datasets. In: IEEE Symposium on Security and Privacy. pp. 111–125.

Neuman, C., Yu, T., Hartman, S., Raeburn, K., July 2005. The kerberos network authentication service (v5). IETF RFC 4120.

OASIS, March 2005. Assertions and protocols for the oasis security assertion markup language (saml) v2.0.
URL http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

OAuth, Aug. 2013. Oauth web site.
URL http://oauth.net

Ohm, P., 2009. Broken promises of privacy: Responding to the surprising failure of anonymization. UCLA Law Review 57, 1701–.
URL http://ssrn.com/abstract=1450006

OpenID, Aug. 2013. Openid website.
URL http://openid.net

OpenID-Foundation, Jan. 2014. Openid connect v.1.0.
URL openid.net/connect/

Park, S., Park, H., Won, Y., Lee, J., Kent, S., Aug. 2009. Traceable anonymous certificate. IETF RFC 5636.

PbD, Aug. 2013. Privacy by design web site.
URL http://www.privacybydesign.ca/

Pereñiguez-Garcia, F., Kambourakis, G., López, R. M., Gritzalis, S., Gómez-Skarmeta, A. F., 2010. Privacy-enhanced fast re-authentication for eap-based next generation network. Computer Communications 33 (14), 1682–1694.

Pereñiguez-Garcia, F., Marin-Lopez, R., Kambourakis, G., Ruiz-Martinez, A., Gritzalis, S., Skarmeta-Gomez, A., 2011. Privakerb: A user privacy framework for kerberos. Computers & Security, 446–463.

Pereñiguez-Garcia, F., Marin-Lopez, R., Kambourakis, G., Ruiz-Martinez, A., Gritzalis, S., Skarmeta-Gomez, A., 2013. Kamu: providing advanced user privacy in kerberos multi-domain scenarios. International Journal of Information Security, 1–21.

Perez-Mendez, A., Pereñiguez-Garcia, F., Marin-Lopez, R., Lopez-Millan, G., 2013. Out-of-band federated authentication for kerberos based on pana. Computer Communications 36 (14), 1527–1538.

Peterson, J., Nov. 2002. A privacy mechanism for the session initiation protocol (sip). IETF RFC 3323.

Pfitzmann, A., Hansen, M., Aug. 2010. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management.
URL http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf

PRISM, 2013. Prism (surveillance program).
URL http://en.wikipedia.org/wiki/PRISM_(surveillance_program)

Ramsdell, B., July 2004. Secure/multipurpose internet mail extensions (s/mime) version 3.1 message specification. IETF RFC 3851.

Ren, J., Wu, J., 2010. Survey on anonymous communications in computer networks. Computer Communications 33 (4), 420–431.

Rescorla, E., Modadugu, N., Jan. 2012. Datagram transport layer security version 1.2. IETF RFC 6347.

Rosenberg, J., Oct. 2009. Obtaining and using globally routable user agent uris (gruus) in the session initiation protocol (sip). IETF RFC 5627.

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E., June 2002. Sip: Session initiation protocol. IETF RFC 3261.

Ruiz-Martinez, A., 2012. A survey on solutions and main free tools for privacy enhancing web communications. Journal of Network and Computer Applications 35 (5), 1473–1492.

Schneier, B., Jan. 2012. Anonymity won't kill the internet.
URL http://www.wired.com/politics/security/commentary/securitymatters/2006/01/70000?currentPage=all

Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V., July 2003. Rtp: A transport protocol for real-time applications. IETF RFC 3550.

Shen, C., Schulzrinne, H. G., 2006. A voip privacy mechanism and its application in voip peering for voice service provider topology and identity hiding. Tech. rep., Department of Computer Science, Columbia University.
URL http://hdl.handle.net/10022/AC:P:29476

Shimaoka, M., Hastings, N., Nielsen, R., July 2008. Memorandum for multi-domain public key infrastructure interoperability. IETF RFC 5217.

Simon, D., Aboba, B., Hurst, R., March 2008. The eap-tls authentication protocol. IETF RFC 5216.

Such, J. M., Espinosa, A., Garcia-Fornes, A., Botti, V., 2011. Partial identities as a foundation for trust and reputation. Engineering Applications of Artificial Intelligence 24 (7), 1128–1136.

Tene, O., 2011. Privacy: The new generations. International Data Privacy Law 1 (1), 15–27.
URL http://idpl.oxfordjournals.org/content/1/1/15.full

TorFone, 2013. Tor fone: p2p secure and anonymous voip tool. V1.1b (01.06.13).
URL http://torfone.org/

Tschofenig, H., July 2010. Federated authentication beyond the web: Problem statement and requirements. IETF ABFAB working group.
URL http://tools.ietf.org/html/draft-tschofenig-moonshot-ps-01

Wasserman, M., Hartman, S., Feb. 2014. Application bridging for federation beyond the web (abfab) trust router protocol. IETF Internet-Draft.
URL http://tools.ietf.org/search/draft-mrw-abfab-trust-router-02

Winter, S., Salowey, J., 2013. Update to the eap applicability statement for abfab. IETF Internet Draft.
URL http://tools.ietf.org/html/draft-ietf-abfab-eapapplicability-06

Wolff, J., 2013. Application-layer design patterns for accountable-anonymous online identities. Telecommunications Policy In Press, Accepted Manuscript, –.

Zhang, G., Fischer-Hubner, S., 2013. A survey on anonymous voice over ip communication: Attacks and defenses. Electronic Commerce Research, –.

Zhu, L., Leach, P., Hartman, S., April 2011. Anonymity support for kerberos. IETF RFC 6112.

Zhu, L., Tung, B., June 2006. Public key cryptography for initial authentication in kerberos (pkinit). IETF RFC 4556.