# ANONYMOUS SERVICES
## *Enhancing End-user Privacy Exploiting Anonymous Networks*

Giovanni Cabiddu, Emanuele Cesena and Davide Vernizzi
*Dip. di Automatica e Informatica, Politecnico di Torino, Torino, Italy*

Keywords: Privacy, Anonymous service, Anonymous network.

Abstract: The large number of online services poses serious problems to users' privacy. The sole confidentiality of data exchanged is not enough for complete privacy because an external observer may learn sensitive information simply by observing the communication channel, even if it is not possible to access the actual data transmitted. In this position paper, we propose a solution where user privacy is guaranteed by providing anonymous access to the services. Our solution is based on a service gateway, an anonymous credential system, an authentication protocol and an anonymous network. We designed the solution to be cost-effective and scalable; moreover, we employ existing standard protocols whenever possible to facilitate development and deployment.

## 1 INTRODUCTION

In the last few years we have faced an increasing diffusion of distributed services. In particular, taking advantage of fast and reliable Internet connections, the paradigm of cloud computing is gradually gaining widespread acceptance. As usage of online services grows, the privacy of the users becomes a big issue. More and more people tend to upload, elaborate and store online personal data such as emails, pictures and contacts, but also appointments and medical data.

In this context, providing user anonymity is the ideal solution for enhancing privacy, but in practice, complete anonymity is difficult to achieve. Protecting the confidentiality of data transmitted on a network channel is fundamental for user privacy, but it is not enough. Indeed, the sole knowledge that the user accesses a particular service might disclose sensitive information. For instance, let us consider a user that repetitively accesses a medical service: an external observer may correlate the repetitive accesses to some kind of medical problem (and possibly may reveal this information to a medical insurance company) even without being able to observe the actual traffic (e.g., due to a protected channel between user and service).

We focus on network communication and in particular on the access to the services by proposing a solution for enhanced user privacy based on anonymous services. We implement our solution basing it on 1) a service gateway, 2) an anonymous credential system 3) an authentication protocol and 4) an anonymous network.

**Related Work.** Basically, to achieve user privacy, two different approaches are possible. The first protects the *source* (i.e. the user), while the latter attempts to protect the *destination* (i.e. the service).

Source privacy protection is usually achieved by means of anonymous networks. A plausible candidate for an anonymous network is Tor (Dingledine et al., 2004) since it is actually available and already used in many applications. In these solutions, the anonymous network is used to hide the user, anonymizing its network connections.

On the other hand, we focus on destination privacy protection by using the anonymous network on the service side, instead of the user side (cf. Sec. 4), implying less requirements on the client's platform. Moreover, anonymity is usually provided by traversing multiple nodes. While this provides anonymity and ensures against a single attacker, it also greatly reduces the bandwidth available, indeed, in networks of this type, the bandwidth available is lesser or equal to the minimal band offered by traversed nodes (Wendolsky et al., 2007). By moving the anonymous network to the services, we can expect the anonymous network to exploit high-bandwidth nodes, therefore leading to higher performance. Furthermore, we provide an access-control feature, both on users and services, which is missing in anonymous networks such as Tor.



Cabiddu G., Cesena E. and Vernizzi D. (2010).
ANONYMOUS SERVICES - Enhancing End-user Privacy Exploiting Anonymous Networks.
In *Proceedings of the International Conference on Security and Cryptography*, pages 221-225
DOI: 10.5220/0003032202210225

There are other works that provide destination protection. To our knowledge, SAGE (Lin et al., 2009) is the one which has the most similar architecture to ours. However, we provide bidirectional communication while SAGE only allows communication from user to service. In addition, we note that in SAGE most of the security lies in a *trusted authority* which may revoke user anonymity, while in our scenario the trusted third party has a more limited role.

Technically speaking, both Tor and SAGE define their own application protocols, while we make use of standard protocols whenever possible. We believe that this approach makes it easier to change the underlying technology, thus increasing resiliency to specific attacks (e.g. eavesdropping by exit nodes in Tor). Moreover, not being bound to a specific technology leads to easier deployment, especially in corporate scenarios. Finally, this choice supports heterogeneous services which are based on different network technology.

In addition, anonymous network, another key component of our solution is the anonymous credential system which is used to provide anonymous access control. These (see, e.g. (Chaum, 1985; Camenisch and Lysyanskaya, 2001)) are cryptographic tools that allow users to obtain credentials from an issuer and can use them to access services, but their communication remains unlinkable even when the services collude with the issuer. Building on such systems, it is possible to extend classical authentication primitives to take into account the privacy aspects of the users. A particular noteworthy anonymous credential system is the *Direct Anonymous Attestation (DAA)* (Brickell et al., 2004): thanks to its flexibility its usage has already been proposed in practical solutions based on Transport Layer Security (TLS) and IPsec (Balfe et al., 2005; Cesena et al., 2010).

**Organisation.** This work is organised as follows: Sec. 2 describes our model and assumptions, while Sec. 3 lists the requirements we want to fulfill. In Sec. 4 we present our solution which implements anonymous services. We conclude in Sec. 5.

A full version of this paper containing a more elaborated discussion of the solution is available at http://security.polito.it/tc/anon-services/.

## 2 MODEL AND ASSUMPTIONS

In this section we present the actors that play a role in our solution, explaining their goals, and we describe the underlying assumptions.

To describe our problem we introduce a model which defines the following participants:

- **Service ($\mathcal{S}_i$).** This is an entity that offers a service. Usually they are provided by companies or organizations (e.g., banks, hospitals).
- **User ($\mathcal{U}$).** This is any authorized entity that wants to access the service $\mathcal{S}_i$.
- **Router ($\mathcal{R}$).** This is an entry point for $\mathcal{U}$ to the Internet. It is controlled by the Internet Service Provider chosen by $\mathcal{U}$. It is capable of seeing all the traffic generated and received by $\mathcal{U}$.
- **Introduction Points (IP).** This is the set of entry points for $\mathcal{U}$ to the service $\mathcal{S}_i$. Each of them allows $\mathcal{S}_i$ to be anonymously accessed, namely without using a direct connection between $\mathcal{U}$ and $\mathcal{S}_i$.
- **Gateway ($\mathcal{G}$).** This is a gateway which notifies services that a user want to access a particular $\mathcal{S}_i$ without disclosing any sensitive information.

The attacker is a third party that is able to read, modify or drop messages between $\mathcal{U}$ and $\mathcal{S}_i$ and collect sensitive information. In addition, it can personify one of the participants. It could be the Service Provider that provides $\mathcal{R}$ or another entity which has physical access to the network.

We base our solution on the following assumptions:

**Anonymous Network (AN).** A network capable of providing anonymous connections to the services is available. In particular this network allows a node to connect to external entities so that no information that may lead to its identification is revealed. Protection of privacy of data exchanged is easily provided by a secure channel. On the other hand, preventing an observer from gaining information by observing the connection is much harder. Therefore, in this context, we do not consider the data exchanged, but we use the anonymous network to protect the information that an external observer may extract by observing the connection.

**Hidden Services.** The anonymous network allows a node inside the network to offer services to external entities anonymously. Using this feature, a service inside AN, can be anonymously contacted by external entities through some Introduction Points (IP).

**Broadcasting.** AN is capable of reliably delivering broadcast messages to its nodes. This feature ensures that any broadcast message sent by an external entity is delivered to the nodes without specific delays that may reveal which node has received the message.

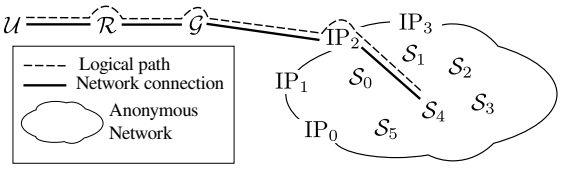


Figure 1: Overview of the anonymous network (AN); a generic user may contact a service using the related Introduction Point (IP).

**Trusted Computing Assumptions.** Nodes that compose the AN can be *trusted platforms*, equipped with a trustworthy Trusted Computing Base (TCB) including a Trusted Platform Module (TPM) (Trusted Computing Group, 2007). The TCB protects critical data and access to authentication credentials from remote attacks. Moreover, the TCB ensures the behaviour of some *trusted components* (e.g., network stack) and is capable of vouching for the integrity of the trusted components to external entities.

An abstract representation of this network is provided in Fig. 1.

# 3 REQUIREMENTS

We want to design a system capable of addressing the following functional and security requirements:

**User Privacy.** A user $\mathcal{U}$ may connect to a certain service $\mathcal{S}_\mathrm{r}$ being ensured that no one, except himself and $\mathcal{S}_\mathrm{r}$, is capable of discovering which service is accessed. This is the central requirement of this work.

**Access Control.** Only authorized services can access the system. This means that a service that did not correctly joined the system must not be able to provide its services (even if it collude with users). This requirement leads to a business model for $\mathcal{G}$ where services pay for joining the system.

**Cost-effectiveness.** The system is designed to be cost-effective. In particular, since a single $\mathcal{G}$ serves multiple users and services, its effort is required to be minimal. This requirement increases the expectations of deploying the system.

**Compliance to existing Standards.** Wherever possible, the system employs standard and diffused protocols. This requirement facilitates the integration and the deployment of the system and makes its development affordable.

**Scalability.** The system is designed to scale well to a large number of users and services. This requirement is crucial since the anonymity level grows with the number of services.

# 4 OUR SOLUTION

This section describes our solution in detail. A discussion on how the security requirements are fulfilled and additional considerations on the feasibility of the solution are presented in the full version of this paper.

Our solution is based on five network protocols that lead to the establishment of an interaction between a user $\mathcal{U}$ and a required remote service $\mathcal{S}_\mathrm{r}$ in a fashion so that no external entity can identify which service is accessed by $\mathcal{U}$.

**Service-initiated Protocols.** Before any interaction with $\mathcal{U}$ may happen, $\mathcal{S}_\mathrm{r}$ must join the system. In order to do this, it runs the following protocol:

- `Join`: This protocol is run between a service $\mathcal{S}_\mathrm{i}$ and $\mathcal{G}$ and results in the admittance of $\mathcal{S}_\mathrm{i}$ to the group of allowed services. During this phase $\mathcal{S}_\mathrm{i}$ fulfils the conditions required by $\mathcal{G}$ (e.g., payment). Moreover, $\mathcal{G}$ and $\mathcal{S}_\mathrm{i}$ set up all the credentials required by the anonymous credential system used. In the case of a DAA-based anonymous credential system, this phase corresponds to the DAA Join protocol.

**User-initiated Protocols.** For accessing the required service $\mathcal{S}_\mathrm{r}$, $\mathcal{U}$ runs the following protocols:

- `Setup`: This protocol is run between $\mathcal{U}$ and the required service $\mathcal{S}_\mathrm{r}$ and results in an agreement between $\mathcal{U}$ and $\mathcal{S}$. The two participants exchange a shared secret that will be used later in the `Wake-up` and the `Authentication` protocol.

  In the shared secret lies most of the privacy of our solution and, therefore, it must be strongly protected; for simplicity, we consider this protocol out-of-band. However, this protocol strongly depends upon the type of service offered by $\mathcal{S}_\mathrm{r}$. For instance, it is possible to conceive an online shared secret distribution (based on a TTP), as long as it still provides the same level of privacy and confidentiality.
- `Wake-up`: This protocol is run among $\mathcal{U}$, $\mathcal{G}$ and $\mathcal{S}$ and results in a notification to $\mathcal{S}_\mathrm{r}$ that $\mathcal{U}$ requires access. During this protocol, $\mathcal{U}$ sends an authentication request for the service $\mathcal{S}_\mathrm{r}$ to $\mathcal{G}$ who will then broadcast the request to the services associated; the correct $\mathcal{S}_\mathrm{r}$ is, hence, informed

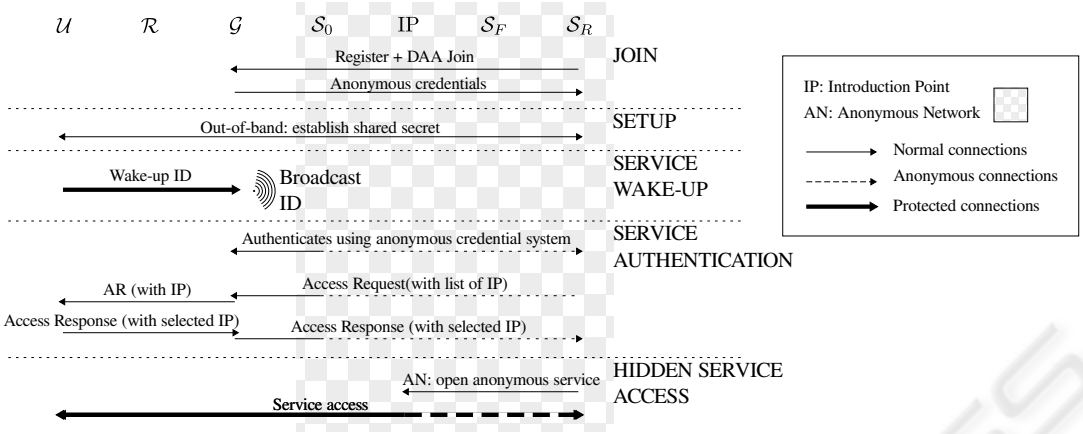


Figure 2: Our solution to provide anonymous services: details on the five protocols involved.

that $\mathcal{U}$ is requiring access and can prepare for the `Authentication` phase. We anticipate that the authentication request sent by $\mathcal{U}$ derives from the shared secret established in the `Setup` phase (e.g., by using hash functions).

- `Authentication`: This protocol is run between $\mathcal{S}_r$ and $\mathcal{U}$ and results in the authentication of the service and in a session key which will be used in the `Access` phase; note that this session key must depend on the shared secret exchanged in the `Setup`.

  In this protocol $\mathcal{G}$ acts as a pass-through entity which forwards messages between $\mathcal{S}_r$ and $\mathcal{U}$. Before allowing $\mathcal{S}_r$ to proceed to the next phase, $\mathcal{G}$ exploits the anonymous credential system to verify that $\mathcal{S}_r$ is one of the services entitled to use the service. During this protocol, $\mathcal{S}_r$ authenticates to $\mathcal{U}$. Moreover, it provides this latter with a list of possible Introduction Points (IP) that can be used to contact $\mathcal{S}_r$ through the anonymous network and $\mathcal{U}$ replies, specifying the chosen IP.

  A common choice for authentication protocol in point-to-point scenarios is EAP: in EAP a supplicant uses an authenticator to access an authentication server. In our case $\mathcal{S}_r$ acts as EAP supplicant, $\mathcal{G}$ as authenticator and $\mathcal{U}$ as authentication server.

- `Access`: In this phase $\mathcal{U}$ connects to $\mathcal{S}_r$ through the IP previously specified. The channel between $\mathcal{U}$ and $\mathcal{S}_r$ is protected with the shared key previously negotiated and is anonymised by the anonymous network.

In Fig. 2 we provide an example of the five protocols where DAA is used as anonymous credential system and EAP is used as authentication.

# 5 CONCLUSIONS

In this paper we present a novel proposal for enhancing user privacy by anonymizing the services that are used by users. Indeed, we claim that an external observer may derive a lot of information about a user by only inspecting his communication, even when this is protected by cryptographic means. We propose a solution which can be adapted to existing technology, indeed, we assume that an anonymous network capable of providing hidden services is present and we build our solution on top of this, but we do not assume any specific anonymous network. Furthermore, we assume that if an anonymous credential system is available, we can provide access control, but, again, we do not require any specific technology.

# REFERENCES

Balfe, S., Lakhani, A. D., and Paterson, K. G. (2005). Securing peer-to-peer networks using Trusted Computing. In *Trusted Computing*, pages 271–298. IEEE Press.

Brickell, E., Camenisch, J., and Chen, L. (2004). Direct Anonymous Attestation. In *CCS'04, 11th ACM Conference on Computer and Communications Security*, pages 132–145. ACM Press.

Camenisch, J. and Lysyanskaya, A. (2001). An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *Advanced in Cryptology – EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 93–118. Springer.

Cesena, E., Löhr, H., Ramunno, G., Sadeghi, A.-R., and Vernizzi, D. (2010). Anonymous authentication with TLS and DAA. In *To appear in TRUST 2010, 3rd International Conference on Trust and Trustworthy Computing*.

Chaum, D. (1985). Security without identification: Transaction systems to make big brother obsolete. *Commun. ACM*, 28(10):1030–1044.

Dingledine, R., Mathewson, N., and Syverson, P. (2004). Tor: The second-generation onion router. In *13th USENIX Security Symposium*, pages 303–320.

Lin, X., Lu, R., Shen, X., Nemoto, Y., and Kato, N. (2009). SAGE: a strong privacy-preserving scheme against global eavesdropping for ehealth systems. *IEEE J. on Sel. Areas Comm.*, 27(4):365–378.

Trusted Computing Group (2007). TPM main specification level 2, version 1.2, revision 103. https://www.trustedcomputinggroup.org/.

Wendolsky, R., Herrmann, D., and Federrath, H. (2007). Performance comparison of low-latency anonymisation services from a user perspective. In *PET 2007, 7th International Symposium on Privacy Enhancing Technologies*, volume 4776 of *LNCS*, pages 233–253. Springer.