Practical Identity Based Broadcast Encryption for Online Social Networks

ESAT/COSIC - KU Leuven and iMinds
Leuven, Belgium
{first.lastname}@esat.kuleuven.be

Abstract. Nowadays Online Social Networks (OSNs) constitute an important and useful communication channel. At the same time, privacy of the information shared is insufficiently protected by coarse-grained privacy preferences. Cryptographic techniques can provide interesting mechanism to protect privacy of users in OSNs. However, this approach faces several issues, such as, OSN provider acceptance, user adoption, key management and usability. We suggest a practical solution that uses Identity Based Encryption (IBE) to simplify key management and enforce confidentiality of data in OSNs. Moreover, we devise an outsider anonymous broadcast IBE scheme to disseminate information among multiple users, even if they are not using the system. Finally, we demonstrate the viability and tolerable overhead of our solution via an open-source prototype.

1 Introduction

Online Social Networks (OSNs), such as Facebook, Google+, and Twitter are increasingly being used and have become a prominent communication channel for many millions of users. OSNs offer users an efficient and reliable channel to distribute and share information. At the same time, OSNs store large amounts of data which prompts several privacy concerns. In particular, it is possible to infer a considerable amount of sensitive information from the shared and stored content. Currently, users are allowed to configure "privacy preferences" in order to limit and select which users or groups can access the shared content. These preferences are generally too coarse-grained and difficult. Another problem is that these preferences do not exclude the provider along with the dangers of data leaks [14], nor external governments [25] and legal subpoenas [23].

All these worrisome issues motivate the need for effective techniques to properly protect user's privacy in OSNs. Several solutions have been proposed and advocated to use cryptographic mechanisms in order to address the privacy issues, either by an add-on atop of existing OSNs [1, 4, 17, 22], or by complete new privacy-friendly architectures [8], mainly decentralized [9, 10]. In general, those solutions suffer from user adoption and key management issues as users are required to register and then share, certify and store public keys. Completely new architectures represent a difficult step for users as the trade off of moving away from the commonly used social ecosystem compared with the risk of losing interactions is high. Arguably, current centralized OSNs are here to stay and will be continue to be actively used by millions of people. In light of motivated by recent events, such as Edward Snowden's whistle-blowing on US surveillance programs [25], OSN providers have all interest in supporting users with extra privacy friendly tools.

Main Idea: Identity Based Encryption (IBE) [29] solutions overcome the key management problem as the public key of the user can be represented by any valid string, such as the email, unique `id` and username, presenting a nice property for OSNs. In addition, savvy and concerned users can always share encrypted content with other users who are not using the the solution thereby motivating the curious ones to use the system as well. Nevertheless, IBE-based systems require a trusted central Private Key Generator (PKG) server to generate the private parameters for each user based on a master secret. Consequently, such an architecture only shifts the trusted party from the OSN to the PKG. However, this problem can be mitigated if the master secret is divided among multiple PKGs following a Distributed Key Generation (DKG) [24] protocol based on Verifiable Secret Sharing (VSS) [7]. A DKG protocol allows n entities to jointly generate a secret requiring that a threshold t of the n entities does not get compromised. In fact, each entity holds only a share of the master secret, that can be reconstructed by at least t shares.

Many OSN users are not only represented on a single OSN but on several, thus, can also hold multiple public keys. Moreover, the multi-PKG setting could be supported and maintained by several existing OSNs. In particular, collaboration between competing OSN providers that compete along is assumed to be a difficult task and orthogonal to their economical business model. Figure 1 depicts an overview example of the proposed model, where a user authenticates to t-PKGs of his choice using, e.g., a similar token as in open id protocols, to retrieve his private key. This action can be performed after the reception of encrypted content as a consequence of user curiosity . The PKG servers can also be represented by governmental entities from different continents, with no incentives to collaborate nor overcome more powerful adversaries using legal measures [23] among at least t-PKGs.

Contribution: In this paper, we propose a novel practical solution that uses IBE with multiple untrusted PKGs atop of current OSNs. We highlight the fact that those multi-PKGs can be supported by several existing OSNs under the business competition assumption, and motivated by the possible attractive incentives towards more privacy concerned audience. Along with the multi-PKG IBE model we devise an IBE broadcast encryption protocol to support multiple recipients. Using a broadcast IBE-based mechanism we allow users to share content with multiple recipients, even if they are not using the system, and, thus, enforce confidentiality of the data. Finally, we implemented our solution on top of the Scramble Firefox extension [4], and show that only a small amount of overhead is required.

degree $t-1$ such that $f(0) = \mathtt{s}$, where the shares s_i are represented by different points on the polynomial. Any entity with t or more shares can reconstruct $f(x)$ using Lagrange interpolation, and subsequently find $\mathtt{s}$. Further, Chor et al. [7] suggested a Verifiable Secret Sharing (VSS) scheme to allow anyone to verify that the right shares are used. The scheme was extended by Feldman [12] and Pedersen [24].

For multiple parties to jointly generate a secret sharing $\mathtt{s}$, all entities are required to participate in a DKG scheme. Each entity i involved generates a different $\mathtt{s}_i$ and $f^i(x)$, and later on distributes and verifies the shares s_{ij}. Hence, a generic DKG does not require a trusted party, as the master secret is computed as the sum of all the polynomials and can only be retrieved by joining t shares. A generic DKG protocol consists of two phases:

`DKG.Setup`: Every entity i generates a random secret $\mathtt{s}_i$ and computes a polynomial of degree $t-1$. The entity i Distributes a valid share s_{ij} over all the other j entities, along with the commitment to the share. Each entity j verifies the shares and computes the new share $s_j = \sum_i s_{ij}$. The master secret is unknown by each party, and composed by the origin point on the sum of all polynomials $f^i(x)$.

`DKG.Reconstruct`: Each entity i broadcasts its share s_i, and with $t \leq n$ shares, one can reconstruct the master secret s.

The DKG protocol is secure assuming that no adversary is able to corrupt at most t parties.

3 Model

We consider that a user u to be a member of an OSN, such as Facebook, Twitter or Google+. We relax the definition from Boyd [6], such that the user u is connected with other users in the same OSN by a friendship relation. Inherently, u aims to interact and share information $\mathtt{m}$ with other users. Each user holds a public and private key pair which is given by an IBE identity server (composed of multiple PKG servers), such that the public key is represented by the `id` of the user in the OSN. Note that each user can be registered into multiple OSNs and hold different public keys and `ids`. We assume that the authentication between the user and the identity servers is done using a token similarly to open id, and performed under an authenticated channel, e.g., TLS.

Threat Model: We consider an adversary to be any entity attempting to passively access the shared information $\mathtt{m}$ by monitoring the sharing channel but with no motivational incentive to tamper with the content. This can be any curious user in the OSN, the OSN provider or even a government agency [25]. Such an adversary should not learn the content nor the identity of the members in the recipient set $\mathcal{S}$, otherwise we consider that the adversary breaks confidentiality and the outsider recipient anonymity of the protocol as defined in [11]. In addition, we assume that such an adversary cannot compromise more than t identity servers. Furthermore, we stress that such adversary cannot control the user computing environment. Also, it is hard to protect against a malicious recipient who copies or forwards shared content. In this case, we say that such recipient breaks the social contract. We stress that we offer no protection against traffic analysis or timing attacks.

Goals: We aim to protect OSN users' privacy by ensuring confidentiality, data integrity and outsider recipient anonymity [11]. In this way we allow users to enforce access control without having to rely on the privacy preferences offered by the OSN. At the same time, we aim at limited modifications to the OSN environment. In particular, we require as little effort as possible and prior knowledge from users in order to achieve a user-friendly scheme as defined by Balsa et al. [2]. In contrast to previous solutions, users are allowed to be in the recipient set by default.

4 Practical IBE for OSNs

In this section, we describe our system. The proposed solution is based on the IBE scheme from Boneh et al. [5] and a relaxed version of the broadcast scheme from Libert et al. [21]. Further, the system relies on a DKG protocol as described by Pedersen [24] to bootstrap multiple PKGs. In addition, we converted the schemes from using Type 1 (i.e., $\mathbb{G}_1 = \mathbb{G}_2$) to Type 3 (i.e., $\mathbb{G}_1 \neq \mathbb{G}_2$) pairings for efficiency [15] and because Type 1 pairings are no longer secure according Joux in [18].

4.1 Basic Scheme

Considering λ to be the security parameter for a security level of l bits, and $\mathcal{S}$ the set of desired recipients u_i with corresponding $\mathtt{id}_i$, such that $\mathcal{S} = \{u_1, .., u_\eta\}$ where $\eta = |\mathcal{S}|$. Let $\mathcal{G}$ be a generator that satisfies the Bilinear Diffie-Helman (BDH) assumption, and $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ the bilinear map such that $e(aP, bQ) = e(P, Q)^{ab}$ for $P \in \mathbb{G}_1, Q \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_q$ as in [5] . In addition, let $\{C, T\} \leftarrow \mathtt{E}_k(M, A)$ be any secure authenticated symmetric encryption that takes plaintext M and header A as input and generates ciphertext C and authentication tag T as output. Similarly, $\{M, T\} \leftarrow \mathtt{D}_k(C, A)$ can be any secure authenticated symmetric decryption that takes ciphertext C and header A as input and generates plaintext M and authentication tag T as output. Our scheme for OSNs is composed by five randomized algorithms: `Setup`, `KeyGen`, `Publish`, and `Retrieve`.

`Setup`(λ, t, n): Outputs the public *params* of the system with respect to the security parameter λ, the number n of PKGs and the threshold t.

1. Run $\mathcal{G}$ on input λ to generate a prime q, two groups $\mathbb{G}_1, \mathbb{G}_2$ of order q, and an admissible bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Choose random generators $P \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$.

5 Practicalities

To demonstrate the viability of our solution, we implemented a proof-of-~~knowledge~~ concept prototype of the distributed identity based broadcast encryption scheme for OSNs.[1] In this section, we discuss the implementation details and the performance results of the cryptographic blocks.

Implementation: For the client component we modified the cryptographic library from Scramble [4] as it is an available open source privacy preserving project. In addition, Scramble is implemented as a Firefox Extension compatible with Firefox 14+, but as it is written in simple Javascript it could easily be ported to other browsers, e.g., Chrome. We implemented the multiple PKG servers in PHP. The bilinear pairing and cryptographic blocks for the PKG and the client component are implemented using the multi-precision MIRACL library [27]. To overcome the limitation of accessing binary code from a browser extension implementation, a local client-server socket implementation was used for the cryptographic requests. For the DKG library we used the available implementation from Kate and Goldberg [20][2] to generate the collective master secret key for the $(n,t)-$PKG servers. AES-GCM [26] was used for the authenticated encryption. For the public key the Facebook username was used, i.e., `id` = *facebook.com/user.name*.

Performance: Experiments were conducted on a Intel Core 2.4 GHz i5 processor with 8 Gb of 1600 MHz DDR3L onboard memory. Table 1 illustrates the execution times for the scheme proposed in Section 4 for $\lambda = 256$bits. Each recipient has to decrypt W_i an average of $\mathcal{O}(n/2)$ times to retrieve the secret and subsequently decrypt the secret message $\mathtt{m}$.

Number of recipients	Encryption time	Decryption time
1	284.529 *ms*	275.409 *ms*
10	2564.510 *ms*	460.930 *ms*
15	3799.600 *ms*	560.611 *ms*
50	12300.500 *ms*	1237.820 *ms*
100	25867.700 *ms*	2260.230 *ms*

Table 1. Execution time of the BE scheme for varying sizes of the recipient set

We also analyzed the execution times of the IBE scheme, as it represents the most costly part of the scheme. Furthermore, our solution uses the random oracle assumption; we show in Table 2 that there is a significant computational difference between a similar scheme in the standard model, i.e., Gentry [16].

[1] Source of our implementations is available upon request.
[2] https://crysp.uwaterloo.ca/software/DKG/ → include this in the bibliography

Nevertheless, we believe that our solution presents a tolerable cost to average users with 100 friends and a usual group size of 15 [30].

	Boneh et al.	Gentry
`IBE.Setup`	368.10 *ms*	424.49 *ms*
`IBE.Extract`	13.84 *ms*	37.46 *ms*
`IBE.Encrypt`	271.90 *ms*	1136.65 *ms*
`IBE.Decrypt`	252.82 *ms*	911.32 *ms*

Table 2. Comparison of execution time for different IBE schemes

6 Related Work

The increased popularity of Online Social Networks (OSNs) and the amount of disseminated information prompted several privacy concerns. Guha et al. [17] proposed NOYB a solution that replaces the personal details of users by fake information. Later, FaceCloak [22] and Scramble [4] make use of cryptographic mechanisms to enforce privacy to the published information. Further, Persona suggest an attribute based encryption scheme for social networks. However, all the aforementioned solutions suffer from a complex key management infrastructure.

Other solutions take a more drastic approach by proposing novel, privacy-friendly architectures meant to replace existing platforms [8, 10, 9]. Besides the privacy protection offered, these solutions face a reduced user wilingness to adopt to a new platform.

Recently, Jung et al. [19] proposed a key management scheme based on dynamical IBE for decentralized OSNs. Their scheme, however, presents several problems. Foremost contains a single point of failure as a trusted party should generate the secret keys for a given `id`. Also, it still requires an additional public key for broadcasting, thus, not solving the key management issue.

In general all previous schemes require public parameters that should be shared and verified by users. In addition, by using an Identity-based scheme we allow users to motivate their friends to use the solution, as registered users can already encrypt messages to unregistered friends.

7 Conclusion

Identity Based Encryption (IBE) solutions provide desirable properties to construct mechanisms to deliver privacy in OSNs. The minimal additional architectural support and the increased ease of key management represent a major motivation to implement IBE in OSNs. We show that using secret sharing and

Naam : Filipe Beato

Promotor : Bart Preneel

Jaarlijkse vakantie	Totaal aantal vakantiedagen
2013	25

Do not send this card anywhere, just keep it in your own desk.

Data		Jaarlijkse vakantie	Klein Verlet	Sociaal Verlof	Reden klein verlet Sociaal verlof	Handtekening Promotor	Saldo
Van	Tot en met	Aantal dagen					25
6-5-2013	14-5-2013	6					19
8-8-2013	8-8-2013	1					18
16-8-2013	16-8-2013	1					17
23-9-2013	1-10-2013	6					11
16-10-2013	16-10-2013	1					10
18-12-2013	20-12-2013	3					7
17-01-2014	17-01-2014	1					6
21-02-2014	21-02-2014	1					5
25-04-2014	29-04-2014	3					2