The Journal of Systems and Software 118 (2016) 36–48

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

A new framework for implementing identity-based cryptosystems

A. Siad [a,*], M. Amara [b]

[a] Research and Development Center CRD Reghaia, Algiers, Algeria
[b] Research and Development Center CRD Reghaia, Algiers, Algeria

ARTICLE INFO

Article history:
Received 28 September 2012
Revised 10 March 2016
Accepted 25 April 2016
Available online 29 April 2016

Keywords:
Cryptographic protocols
Distributed protocols
Software implementation

ABSTRACT

Identity-Based Encryption (IBE) suffers from the problem of trust in the Private Key Generator (PKG), which translates into the ability of the PKG to produce and distribute multiple private keys or multiple copies of a single key without knowledge of the genuine user. This problem makes the deployment of these systems limited to areas where trust in the PKG must have a high level. Many works addressed this problem and proposed a wide range of key generation protocols which grew from simple protocols run between user and PKG to complex and interactive protocols involving distributed computations. However, the implementation of such complex protocols requires much programming efforts and the few existing tools and libraries deal with special case protocols.

In this paper, we present the first complete, efficient and modular framework, composed of a set of libraries, which brings together the most known techniques of private-key generation for identity-based cryptosystems. Our framework aims at providing robust tools designed in a modular and reusable manner to allow developers to implement the latest results coming from theoretical cryptography.

1. Introduction

The fundamental concern of information assurance today is the issue of trust. On this basis, Public Key Infrastructures (PKI) were used successfully to manage trust between entities and provide different mechanisms for issuing, maintaining, and revoking certificates. However, these operations on certificates have made PKIs increasingly complex to manage and to deploy in practice. Simplifying the management of PKI by removing certificates requires new mechanisms to reduce the trust in public keys. This problem was one of the major challenges of modern cryptography.[1]

In 1984, Shamir (1984), solved this challenge by presenting a novel concept called IBE. The idea is to use a scheme in which the user's identity (name, email address, IP address, etc.) is used as public key. The user's private key is derived from the public key by a trusted third party called private key generator (PKG) and delivered to the user via a secure channel.

It is clear that IBE has successfully removed the need for certificates. However, despite the benefits of not having to manage a PKI, IBE introduced a problem of trust in the PKG commonly known as key escrow problem. Consequently, the PKG is able to generate private keys without the knowledge of the genuine user and proceed to decrypt messages, or worse, generate and distribute keys to other potentially malicious users. For this reason, the users must have a very high level of trust on the PKG, which may be undesirable in many application scenarios.

An important issue and a natural question to ask is: "how can we reduce the trust one should have in the PKG?". This question was asked by authors of the first efficient IBE scheme (Boneh and Franklin, 2001), and it persists until nowadays. The answer is simple but its implementation is difficult, it consists of designing key generation and extraction protocols to reduce this trust. These protocols were the subject of a flurry of research (Boneh and Franklin, 2001; Chen et al., 2002; Gentry, 2003; Al-Riyami and Paterson, 2003; Lee et al., 2004; 2005; Green and Hohenberger, 2007; Goyal, 2007; Goyal et al., 2008; Libert and Vergnaud, 2009; Camenisch et al., 2009; Chow, 2009; Geisler and Smart, 2009; Abdalla et al., 2009; Boyen et al., 2010; Kate and Goldberg, 2010; Lee, 2010; Sahai and Seyalioglu, 2011; Siad and Amara, 2011; Siad, 2012a; 2012b), where several solutions were presented.

Following this theoretical development, a small number of libraries and frameworks (Waters, 2006; Lynn, 2002; Scott, 2010; Akinyele et al., 2011; 2013) were developed and proposed a limited set of tools for implementing identity-based cryptosystems. On one hand, IBE schemes require key generation protocols which

* Corresponding author.
E-mail addresses: siad@math.univ-paris13.fr (A. Siad), amara.moncef@yahoo.fr (M. Amara).

[1] Part of this work was done within Geometry and applications Laboratory at Paris 13 University-France and also the cryptography team at Paris 8 University-France, when first Author was Ph.D Student.

http://dx.doi.org/10.1016/j.jss.2016.04.059
0164-1212/

grew from simple algorithms to interactive and complex protocols involving distributed computations. However, there is no existing library which provides support to implement these protocols, in particular the distributed ones. On the other hand, implementation of identity-based cryptosystems requires extensive knowledge in software design and programming.

The original motivation of this work is to design a new framework bringing together different techniques of key generation for IBE schemes including various sub-protocols needed such as verifiable secret sharing, distributed key generation, and Zero Knowledge proof protocols. At the very beginning of this work, we set two main goals expected from our implementation. On one hand, show that it is possible to implement in practice protocols, in particular distributed protocols, proven secure in different security models while maintaining simplicity, efficiency, and reusability of sub-protocols in other protocols or even in larger and complex protocols. On the other hand, consider a possible practical use of our framework and we project that the framework is also interoperable with other systems.

Our contribution. The contributions of this paper can be summarized as follows: First, we draw a state of the art of existing key generation protocols along with practical implementations in the field of identity-based encryption. We classify these protocols into several main approaches, each one containing a set of subclasses, and we highlight the need for a new framework to allow easy implementation and rapid prototyping of such protocols. To the best of our knowledge, this is the first framework[2] that provides tools for implementing key generation protocols for IBE in particular the distributed protocols. Then, we establish a set of design criteria to be met by our framework and we propose a new design approach which consists of fully specifying the framework in an abstract form. Our approach of design combines the object-oriented method and the method based on layers. Finally, we conduct extensive experiments to evaluate the performance of different modules composing the framework for different parameters. We end-up with an analysis of the obtained results showing that our modules meet the goals fixed at the very beginning stages of this work.

Outline of the paper. The remainder of this paper is organized as follows: in Section 2 we review existing works (cryptographic protocols and libraries). Section 3 will be dedicated to the presentation of technical indispensable preliminaries and definition. In Sections 4 and 5, we present our approach of design and software architecture. In Section 6 we describe our Benchmark and give experimental results which we analyse in depth. Finally, Section 7 draws a conclusion and discusses future work.

2. Related work

2.1. Key generation protocols

The problem of trust in the PKG makes the use of identity-based cryptosystems very limited. To extend these cryptosystems to be used by a large community of users and developers, various protocols for private key generation have been studied. We use similar terminology to the one used in almost all papers, and we classify these protocols into two categories: the blaming approach and the preventive approach. In the first approach the PKG can be blamed in case of malicious behaviour. This approach groups accountable protocols. The second approach aims at minimizing the chance that the PKG harms an honest user. In this approach, we distinguish three subclasses: distributed protocols, blind and anonymous key-issuing protocols, and protocols based on secret information.[3]

2.1.1. IBE with distributed PKG

Threshold cryptosystems give ways to distribute trust throughout a group and increase the availability of cryptographic systems. A standard approach in designing these protocols is to use secure multi-party computation to distribute algorithms. However, general multi-party computation cannot be directly applied without incurring a considerable computation penalty. In the IBE settings, Boneh and Franklin (2001), were the first to propose to split the master secret key over multiple $\mathcal{PKG}$s, the user obtains a partial private key from each $\mathcal{PKG}$ and reconstructs her private key in threshold manner. Boneh et al. (2006), presented threshold IBE which is then transformed generically to a chosen ciphertext secure public key threshold encryption system. Geisler and Smart (2009), proposed a distributed version of Sakai-Kasahara based systems. The authors presented a solution which requires for each ID-based key a secure multi-party computation to be performed amongst the servers and presented an implementation which is more efficient than general multi-party computation protocols. In 2010, Kate and Goldberg (2010), developed a distributed private key generators for three IBE schemes along with the security proofs in the random oracle model. The authors proposed as an application the Onion Routing protocols. Abdalla et al. (2009), and Boyen et al. (2010), take a completely different approach and propose to replace $\mathcal{PKG}$ by a distributed set of users, each one of them holding a small piece of the master secret in the form of a short password. Recently, Siad (2012a), proposed a generic transformation of IBE schemes into IBE schemes with distributed PKG under the standard definitions of security following the ideal/real model paradigm (Canetti, 2000).

2.1.2. Blind IBE and anonymous key issuing (AKI)

This class of protocols uses identity blinding techniques when requesting a private key from the PKG, the latter generates private keys for users that he does not know the identity. These protocols usually require special infrastructure consisting of entities providing additional registration and certification of user identities. In this context, Green and Hohenberger (2007), introduced the notion of blind IBE which consists of a IBE scheme with a blind protocol BlindExtract for extracting private keys. Camenisch et al. (2009), extended blind IBE to committed blind anonymous IBE that allows key extraction on committed identities and developed a scheme based on Boyen-Waters IBE (Boyen and Waters, 2006). Chow (2009), proposed the first anonymous key issuing protocol (AKI) for a modified version of Gentry scheme (Gentry, 2006). The author applied the security properties of p-signature schemes to his protocol and proposed a practical architecture for the deployment of such protocols by introducing a new entity called identity certification authority. The latter should not reveal users' identities to the PKG. Siad and Amara (2011), combined this class of protocols with distributed protocols to construct an anonymous key issuing protocol with a distributed PKG.

2.1.3. Accountable IBE

It is clear that in the case of preventive approach, [4] the PKG can decrypt all messages in the system, even in the case of AKI protocols the PKG can use a brute-force search, or worse, collaborate

[2] While we are inspired by the modular architecture developed by Akinyele et al. (2011, 2013), our framework has the advantage of being more specialized since we focus only on key generation protocols, particularly the distributed ones, for identity-based cryptosystems. Furthermore, we made the efficiency of protocols as our first priority by choosing the C++ language.

[3] This subclass is composed of certificateless cryptography (Al-Riyami and Paterson, 2003), and certificate-based cryptography (Gentry, 2003), which we don't implement since they are considered non-identity based solutions.

[4] Exception made for distributed protocols. However, nothing prevents a coalition of PKG to cooperate for such a capability.