AN EFFICIENT SECURE AUTHENTICATION USING IDENTITY BASED CRYPTOGRAPHY IN TWO-SERVER PAKE PROTOCOL

Nimmi George¹ and George T. Vadakkumcheril²

¹ P.G Scholar, Department of CSE, K.M.P College of Engineering, Odakkali, Kerala, India
² Asst. Professor, Department of CSE, K.M.P College of Engineering, Odakkali, Kerala, India

Abstract— Security is an important issue in the client-server communication. Password is the basic function used for the client-server communication process. Ubiquitous information access and communication has become essential in daily life and global enterprises. In two-server password-authenticated key exchange (PAKE) protocol, a client splits its password and stores two shares of its password in the two servers, respectively, and the two servers then cooperate to authenticate the client without knowing the password of the client. In case one server is compromised by an adversary, the password of the client is required to remain secure. In this paper, we present two compilers that transform any two-party PAKE protocol to a two-server PAKE protocol on the basis of the identity-based cryptography, called ID2S PAKE protocol. By the compilers, we can construct ID2S PAKE protocols which achieve implicit authentication. As long as the underlying two-party PAKE protocol and identity-based encryption or signature scheme have provable security without random oracles, the ID2S PAKE protocols constructed by the compilers can be proven to be secure without random oracles. We initiate the secure process by verifying two-party protocol to two-server protocols on the basis of identity based cryptography. Authentication is the main factor of the project. Experimental result proves the effectiveness of the system.

Index Terms— Client server communication, Ubiquitous information, Password systems, Identity cryptography and Authentication.

I. INTRODUCTION

UBIQUITOUS information access and communication have become essential to everyday life, global business, and national security. Activities, including personal, commercial and international financial transactions, studying and teaching, shopping for goods or managing modern battlefields have fundamentally changed over the last decade as a result of the expanding capabilities of computers and networks [1]. Most such activities are supported by distributed applications which, in turn, increasingly rely on messaging systems to provide secure and uninterrupted service within acceptable throughput and latency parameters. This is difficult to guarantee in a complex network environment that is susceptible to a multitude of human and/or electronic threats, especially, as network attacks have become more sophisticated and harder to contain [2].

Password-authenticated key agreement generally encompasses methods such as:

1. Balanced password-authenticated key exchange
2. Augmented password-authenticated key exchange
3. Password-authenticated key retrieval
4. Multi-server methods
5. Multi-party methods

In the most stringent password-only security models, there is no requirement for the user of the method to remember any secret or public data other than the password.

A distributed messaging system is essentially an abstraction layer built on top of an underlying network. It provides distributed applications with: (1) services not available from the native network, e.g., security, ordered message delivery, or (2) services that are enhanced, e.g., higher availability, improved reliable delivery. Group communication systems, overlay networks, and middleware are all examples of messaging systems serving as infrastructure for applications, such as: web clusters, replicated databases, scalable chat services and streaming video [3]. Since many applications are expected to run over the Internet, security becomes a real necessity. We note that even for applications running in local area networks, particularly in commercial environments, security is required to ensure restricted access to data and to protect communication according to regulations and hierarchical structures specific to an organization. Although not an independent service, security [4] is an enabling feature without which the actual end-services cannot be trusted or relied upon.

The rest of the paper is organized as follows: Section II describes the related works; Section III depicts the proposed framework; Section IV depicts the performance of the results and at last concludes in Section V.

II. RELATED WORK

The first successful password-authenticated key agreement
methods were Encrypted Key Exchange methods described by Steven M. Bellovin and Michael Merritt in 1992. Although several of the first methods were flawed, the surviving and enhanced forms of EKE effectively amplify a shared password into a shared key, which can then be used for encryption and/or message authentication. The first provably-secure PAKE protocols were given in work by M. Bellare, D. Pointcheval, and P. Rogaway (Eurocrypt 2000) and V. Boyko, P. MacKenzie, and S. Patel (Eurocrypt 2000). These protocols were proven secure in the so-called random oracle model (or even stronger variants), and the first protocols proven secure under standard assumptions were those of O. Goldreich and Y. Lindell (Crypto 2001) which serves as a plausibility proof but is not efficient, and J. Katz, R. Ostrovsky, and M. Yung (Eurocrypt 2001) which is practical.

The first password-authenticated key retrieval methods were described by Ford and Kaliski in 2000.

Research in group communication systems operating in a local area network (LAN) environment has been quite active in the last 15-20 years. Initially, high availability and fault tolerance were the main goals. This resulted in systems like ISIS [12], Transis [13], Horus [14], Totem [15], and RMP [16]. These systems explored several different models of group communication such as Virtual Synchrony [3] and Extended Virtual Synchrony [4]. More recent work in this area focuses on scaling group membership to wide area networks (WAN) [17], [18]. With the increased use of GCS-s over insecure open networks, some research interests shifted to securing these systems. Research on securing group communication is fairly new. The only implemented GCS-s that focus on security (in addition to ours) is: the SecureRing [19] system at UCSB, the Horus/Ensemble systems at Cornell [20], [21], and the Rampart system at AT&T [22].

At the core of any GCS is a membership protocol. Some of the work in securing group communication focused on protecting the membership protocol in the presence of Byzantine faults. This includes systems such as Rampart [22] and SecureRing [19]. Rampart builds its group multicast over a secure group membership protocol achieved via two-party secure channels. SecureRing protects its low-level ring protocol by using digital signatures to authenticate each token transmission and each data message received. Both systems exhibit limited performance since they use relatively costly protocols and make extensive use of public key cryptography.

In addition to the membership service, GCS-s provide reliable ordered message delivery within a group. To secure this service, group members (senders) must be authenticated and both confidentiality and integrity of client data must be guaranteed. One notable work in this area is the Horus/Ensemble work at Cornell [23], [20], [21]. Ensemble achieves data confidentiality by using a shared group key obtained via group key distribution protocols. Although efficient, this method does not provide certain security properties such as key independence and perfect forward secrecy. For authentication, Ensemble uses the popular PGP [24] method. In addition, the system allows application-dependent trust models in the form of access control lists which are treated as replicated data within a group. Recent research on Bimodal-Multicast, Gossip-based protocols [25] and the Spinglass system has largely focused on increasing scalability and stability of reliable group communication services in more hostile environments – such as wide-area and lossy networks – by providing probabilistic guarantees about delivery, reliability, and membership.

III. PROPOSED FRAMEWORK

A. Existing System

In the single-server setting, all the passwords necessary to authenticate clients are stored in a single server. If the server is compromised, due to, for example, hacking or even insider attacks, passwords stored in the server are all disclosed. This is also true to Kerberos, where a user authenticates against the authentication server with his username and password and obtains a token to authenticate against the service server.

1) Disadvantages of Existing System

- In PAKE, where two parties, based only on their knowledge of a password, establish a cryptographic key by exchange of messages.
- A PAKE protocol has to be immune to on-line and off-line dictionary attacks. In an off-line dictionary attack, an adversary exhaustively tries all possible passwords in a dictionary in order to determine the password of the client on the basis of the exchanged messages.
- In on-line dictionary attack, an adversary simply attempts to login repeatedly, trying each possible password. By cryptographic means only, none of PAKE protocols can prevent on-line dictionary attacks. But on-line attacks can be stopped simply by setting a threshold to the number of login failures.

B. Proposed System

In this paper, we propose a new compiler for ID2S PAKE protocol based on any identity-based signature scheme (IBS), such as the Paterson et al.’s scheme. The basic idea is: The client splits its password into two shares and each server keeps one share of the password in addition to a private key related to its identity for signing. In key exchange, each server sends the client its public key for encryption with its identity-based signature on it. The signature can be verified by the client on the basis of the identity of the server. If the signature is genuine, the client submits to the server one share of the password encrypted with the public key of the server. With the decryption keys, both servers can derive the same one-time password, by which the two servers can run a two-party PAKE protocol to authenticate the client.

1) Advantages of Proposed System

We have implemented our ID2S PAKE protocols, it shows that our protocols save from 22% to 66% of computation in each server, compared with the Katz et al.’s protocol. The server performance is critical to the performance of the whole protocol when the servers provide services to a great number of clients concurrently.

Our Protocol shows that less than one second is needed for the client to execute our protocols. The system architecture is shown in Fig 3.1.
In the real world, a protocol determines how users behave in response to input from their environments. In the formal model, these inputs are provided by the adversary. Each user is assumed to be able to execute the protocol multiple times (possibly concurrently) with different partners. This is modeled by allowing each user to have unlimited number of instances with which to execute the protocol.

This section explains about the proposed Password-Authenticated Key Replacement Protocols (PAKE) that enhances the security of the client server system. Identity based cryptography is used for ensuring the authentication of the clients. The proposed work is executed as follows:

a) Client Module
b) Server module
c) Private Key Generator (PKG) Module
d) ID2S PAKE system

a) Client Module

In this module, a group of clients are predefined in the networks. The client sends requests to the servers in the public networks. In accord to client, password is the only thing considered for protecting their account. The unique feature is that the password is splitted into two shares and stored in two servers. The client’s identity is taken as input for further verification process.

b) Server module

At first, the groups of servers are initiated. Each server provides services to the clients on the networks. Each client is supplied with private key. The server transmits public key to its clients using identity based signature. It is further verified by the client’s identity. If the signature is genuine, the client submits to the server one share of the password encrypted with the public key of the server. With the decryption keys, both servers can derive the same one-time password, by which the two servers can run a two-party PAKE protocol to authenticate the client.

c) Private Key Generator (PKG) module

The task of Private Key Generator is to generate the private key based on the public parameters for the servers. In case of Identity based Cryptography, the decryption key is obtained from PKG system. The messages are decrypted using the identity of the server. Then, the threshold cryptography is used in master-server key systems. The intention of this study is to secure the keys from any misbehaving practices. In this structure, the signing key is only known to the servers that make the system more secured.

d) ID2S PAKE Module

In this module, we present two compliers system namely, two-party PAKE P to ID2S PAKE P’ using Identity based cryptography. The first execution is based on Identity based Signature (IBS) and the second is Identity based Encryption (IBE).

i) Identity based Signature (IBS): It is the basic building block of the secured system. The compiler consists of two servers A and B and client C with two authenticated keys. If the authentication part is removed from the system, it operates like Diffie Hellman Key Exchange protocol.

ii) Identity based Encryption (IBE): A high-level description of our compiler based on identity based encryption (IBE). The client C (knowing its password pwC) runs the protocol P’ with the two servers A and B to establish two session keys respectively.
IV. EXPERIMENTAL DESIGNS
This section explains about the experimental results of the proposed system.
V. CONCLUSION

The main focus of this work was designing high performance security architecture for a client-server group communication system. In this paper, we propose a novel Password Authenticated Key Exchange Protocol (PAKE) that heightens the security of the client server systems. We initiate the secure process by verifying two-party protocol to two-server protocols on the basis of identity based cryptography. The experimental results we present demonstrate the increased scalability of integrated approaches over layered approaches, without a significant decrease in throughput performance.

REFERENCES