**ID2S Password-Authenticated Key Exchange Protocols**

**ABSTRACT**

Password authenticated key exchange (PAKE) is where two or more parties, based only on their knowledge of a password, establish a cryptographic key using an exchange of messages, such that an unauthorized party (one who controls the communication channel but does not possess the password) cannot participate in the method and is constrained as much as possible from brute force guessing the password. (The optimal case yields exactly one guess per run exchange.) Two forms of PAKE are Balanced and Augmented methods.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. Compared with the Katz et al.’s two-server PAKE protocol with provable security without random oracles, our ID2S PAKE protocol can save from 22% to 66% of computation in each server. Identity-based systems allow any party to generate a public key from a known identity value such as an ASCII string. A trusted third party, called the [Private Key Generator](https://en.wikipedia.org/w/index.php?title=Private_Key_Generator&action=edit&redlink=1) (PKG), generates the corresponding private keys. To operate, the PKG first publishes a master public key, and retains the corresponding master private key (referred to as *master key*). Given the master public key, any party can compute a public key corresponding to the identity *ID* by combining the master public key with the identity value. To obtain a corresponding private key, the party authorized to use the identity *ID* contacts the PKG, which uses the master private key to generate the private key for identity *ID*.



**Existing System**

Earlier password-based authentication systems transmitted a cryptographic hash of the password over a public channel which makes the hash value accessible to an attacker. When this is done, and it is very common, the attacker can work offline, rapidly testing possible passwords against the true password’s hash value. Studies have consistently shown that a large fraction of user-chosen passwords are readily guessed automatically.

**Disadvantage:**

1.The hash value accessible to an attacker.

2.The attacker can work offline, rapidly testing possible passwords against the true password’s hash value.

3.An adversary can always succeed by trying all passwords one-by-one in an on-line impersonation attack. A protocol is secure if this is the best an adversary can do. The on-line attacks correspond to Send queries.

**Proposed System:**

Typical examples are the “encrypted key exchange” (EKE) protocols given by Bellovin and Merritt, where two parties, who share a password, exchange messages encrypted by the password, and establish a common secret key. The formal model of security for PAKE was firstly Based on the security model, PAKE protocols have been proposed and proved to be secure.

A security model for ID2S PAKE protocol was given and a compiler that transforms any two-party PAKE protocol to an ID2S PAKE protocol was proposed on the basis of the Cramer-Shoup public key encryption scheme and any identity-based encryption scheme, such as the Waters’ scheme.

The second model is called password-only model. Bellovin and Merritt were the first to consider authentication based on password only, and introduced a set of so-called “encrypted key exchange” protocols, where the password is used as a secret key to encrypt random numbers for key exchange purpose. Formal models of security for the password-only authentication were first

given independently by Bellare et al. and Boyko et al.. Katz et al. were the first to give a password-only authentication protocol which is both practical and provably secure under standard cryptographic assumption.

Advantages:

1.Establish a cryptographic key for secure communications after authentication.

2.The sense that an adversary attacking the system cannot determine session keys with advantage non-negligibly greater than that of an online dictionary attack.

Modules Description

We present two compilers transforming any two-party PAKE protocol P to an ID2S PAKE protocol P0 with identity-based cryptography. The first compiler is built on identity-based signature (IBS) and the second compiler is based on identity-based encryption (IBE).

1.ID2S PAKE Based on IBS

We need an identity-based signature scheme (IBS) as our cryptographic building block. A high-level description of our compiler in which the client C and two servers A and B establish two authenticated keys, respectively. If we remove authentication elements from our compiler, our key exchange protocol is essentially the Diffie-Hellman key exchange protocol. We present the protocol by describing initialization and execution.

The Diffie-Hellman key exchange protocol was invented by Diffie and Hellman in 1976. It was the first practical method for two users to establish a shared secret key over an unprotected communications channel. Although it is a non authenticated key exchange protocol, it provides the basis for a variety of authenticated protocols. Diffie-Hellman key exchange protocol was followed shortly afterward by RSA, the first practical public key cryptosystem.

Key Generation: On input the identity S of a server S 2 Server, paramsIBS, and the secret sharing master-keyIBS, PKGs cooperate to run ExtractIBS of the IBS scheme and generate a private (signing) key for S, denoted as dS, in a manner that any coalition of PKGs cannot determine dS as long as one of the PKGs is honest to follow the protocol.



2. ID2S PAKE Based on IBE

A high-level description of our compiler based on identitybased encryption. We present the protocol by describing initialization and execution.

Key Generation: On input the identity S of a server S 2 Server, paramsIBE, and the secret sharing master-keyIBE, PKGs cooperate to run ExtractIBE of the IBE scheme and generate a private (decryption) key for S, denoted as dS, in a manner that any coalition of PKGs cannot determine dS as long as one of the PKGs is honest to follow the protocol.

Each user has a private key x

Each user has three public keys: prime modulus p, generator g and public Y = gxmod p

Security is based on the difficulty of DLP

Secure key size > 1024 bits ( today even 2048 bits)

Elgamal is quite slow, it is used mainly for key authentication protocols

Protocol Execution. Given a triple (C; A;B) 2 Client ServerTriple, the client C (knowing its password pwC) runs the protocol P0 with the two servers A (knowing GpwC;A , gpwC;A and its private key dA) and B (knowing GpwC;B , gpw C;B and its private key dB) to establish two session keys, respectively.

 At first, the client randomly chooses pw1 from Zn and computes pw2 = pwC 􀀀 pw1(mod n). Next the client C randomly generates a one-time public and private key pair (pk; sk) for the public key encryption scheme E, and randomly chooses an integer rc from Zq and computes Wc = grc ; h = H1(C;Wc; pk): Next, according to the identities of the two servers A and B, the client C performs the identity-based encryptions Ea = IBE(Gpw1h􀀀1 ;A);Eb = IBE(Gpw2h􀀀1 ;B): Then, the client sends msg1 = hC;Wc; pk;Eai and msg2 = hC;Wc; pk;Ebi to the two servers A and B, respectively.



3.Initialization

The two peer servers S1 and S2 jointly choose a cyclic group G of large prime order q with a generator g1 and a secure hash function H : {0; 1}\*->Zq, which maps a message of arbitrary length into an l-bit integer, where l= log2 q. Next, S1 randomly chooses an integer s1 from Zq and S2 randomly chooses an integer s2 from Zq , and S1 and S2 exchange g1s1 and g1s2 . After that, S1 and S2 jointly publish public system parameters G; q; g1; g2;H where g2 = gs1s2 .

4.Registration

The two secure channels are necessary for all two server PAKE protocols, where a password is split into two parts, which are securely distributed to the two servers, respectively, during registration. Although we refer to the concept of public key cryptosystem, the encryption key of one server should be unknown to another server and the client needs to remember a password only after registration.

# System Configuration

# H/W System Configuration:

#  Processor - Pentium –III

Speed - 1.1 Ghz

RAM - 256 MB(min)

Hard Disk - 20 GB

Floppy Drive - 1.44 MB

Key Board - Standard Windows Keyboard

Mouse - Two or Three Button Mouse

Monitor - SVGA

# S/W System Configuration:

* Operating System :Windows95/98/2000/XP
* Application Server : Tomcat5.0/6.X
* Front End : HTML, Java, Jsp
* Scripts : JavaScript.
* Server side Script : Java Server Pages.
* Database : Mysql
* Database Connectivity : JDBC.

CONCLUSION

We present two efficient compilers to transform any two-party PAKE protocol to an ID2S PAKE protocol with identity-based cryptography. In addition, we have provided a rigorous proof of security for our compilers without random oracle. Our compilers are in particular suitable for the applications of password-based authentication where an identity-based system has already established. Our future work is to construct an identity-based multipleserver PAKE protocol with any two-party PAKE protocol.