

An ID-based Proxy Authentication Protocol Supporting Public Key Infrastructure

Shiuh-Pyng Shieh, Shih-I Huang and Fu-Shen Ho

Department of Computer Science and Information Engineering,

National Chiao-Tung University, Hsinchu, Taiwan 30010

Tel: +886-3-5731876

Fax: +886-3-5724176

Email: ssp@csie.nctu.edu.tw

ABSTRACT

The advantage of the ID-based authentication protocols over public-key based protocols is that authentication can be performed by simply knowing the identity of a party. Meanwhile, Public Key Infrastructure (PKI) provides a suite of excellent security and user management mechanisms that can be easily deployed to the Internet. In this paper, we present an ID-based proxy authentication protocol that can be interoperable with PKI. The proposed protocol leverages the management mechanisms of PKI while inheriting the nature of traditional ID-based protocols. In our protocol, a proxy certificate authority (PCA) is proposed to act as a bridge between an ID-based domain and the PKI domain. Authentication between two entities of different domains is thus made possible with the help of the proxy CA. The proposed protocol minimizes the message exchange overhead within an ID-based domain and supports both initial and subsequent authentication. In addition, a security analysis is presented to verify the strength and efficiency of the proposed protocol.

KEYWORDS

Public Key Infrastructure, ID-based protocol, authentication protocol, proxy authentication.

1. Introduction

An authenticated key exchange protocol is to provide communication parties to know each other's identity and to share a common key known only to them[3]. Generally, there are two types of schemes that support secure and private communication. One is centralized, where a key information center (KIC) is involved to authenticate communication parties and generate a common key. The other is decentralized, which does not require any key information center to deal with the authentication and key distribution. Two of the famous decentralized schemes are the RSA scheme[11] and Diffie et al's public key distribution system[4]. A famous type of scheme with decentralized approaches is called ID-based scheme.[8][9][12][13][14] In ID-based schemes, since authentication does not depend on the public key of a user, there is no need to keep public keys for authentication purpose.

A Public Key Infrastructure (PKI)[1][6][7][10][15] is a key management environment for public key information of a public key cryptographic system [16]. A fundamental element of PKI is a data structure called a certificate, which is used to bind a specific identity to a specific public key and information on how the public key can be used. The most widely deployed certificate specification so far can be found in the International Telecommunications Union X.509 standard [6]. A Certificate Authority (CA) is a trusted third party that issues certificates to users within its administrative domain and provides status information about the certificates that it has issued.

However, PKI does provide an excellent management mechanism for both users and certificates. Traditional ID-based domain users do not know the existence of other users and the expiration status of the other users, because the KIC does not record domain users' information. However, with the help of PKI, it does provide domain user with a management protocol for querying the status of the other domain users.

Many different authentication protocols have been proposed according to different purposes and different domains. As long as the users stay in the ID-based domain and register themselves to the KIC, secure communication using ID-based authentication and key exchange protocols can keep on working without any problem. While the users are in PKI domain, the authentication and key exchange protocols can work only when each user's certificate is still valid. During PKI authentication and key exchange procedure, the CA plays a very important role of checking the validity of each user's certificate. If one user's certificate is invalid, the authentication and key exchange process cannot keep the way they are. Moreover, ID-based domain uses decentralized authentication protocol, while PKI domain uses centralized authentication protocol. In order to support PKI infrastructure, we proposed an ID-based proxy authentication protocol that can cooperate with PKI.

In this paper, we present an ID-based proxy authentication protocol that can be interoperable with PKI. The proposed protocol leverages the management

* This work is supported in part by the National Science Council, Ministry of Education, and Lee & MTI Center, NCTU.

mechanisms of PKI while inheriting the nature of traditional ID-based protocols. In our protocol, a proxy certificate authority (PCA) is proposed to act as a bridge between the ID-based domain and the PKI domain. Authentication between two entities of different domains is thus made possible with the help of the proxy CA.

This paper is organized as follows. In Section 2, we propose a PKI interoperable authentication architecture that is based on ID-based approaches. In Section 3, we present the design of an ID-based authenticated key exchange protocol that is based on the interoperable architecture in Section 2. A security analysis of the proposed protocol and the conclusion are given in Section 4 and 5, respectively.

2. Proposed ID-based Proxy Authentication Architecture

The ID-based authentication protocol has the property that whenever a user registers himself to the KIC, he can securely communicate with the other registered users using separate and individual session keys. ID-based domain users do not need the KIC to join authentication and key exchange procedures. While PKI domain needs the CA to check the validate status of each user's certificate to complete authentication procedure. When users are in the same domain, the authentication and key exchange protocols can both work well. However, when users are in different domain, the authentication and key exchange protocols are more complex than users are only in the same domain. For this reason, we proposed ID-based authentication and key exchange protocols that can support public key infrastructure. By using the proposed protocol, every involved user can authenticate the others under different domain and proceed with key exchange to get a common session key to proceed secure communicate.

The proxy CA (PCA) is the main element of this architecture. Three objects are included in a PCA, that is, a CA in PKI domain, a KIC and a private database in ID-based domain. Moreover, in order to cooperate with these two domains, the KIC stores the CA's ID to be further used in ID-based domain. Figure 1 shows the ID-based proxy CA architecture.

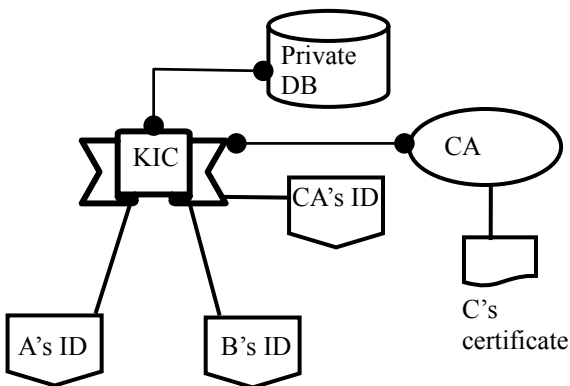


Figure 1 ID-based proxy CA architecture

In ID-based proxy authentication protocol, it needs the following phases to complete the authentication between ID-based domain users and PKI domain users.

A. Initialization Phase

As the system initializes, in order to keep the consistency of both domain, the CA will generate its ID and register it to the KIC. The KIC will store the CA's ID to be used for ID-based domain. Figure 2 shows the components of the proxy CA.

In KIC's point of view, whenever a new user joins ID-based domain, he has to register himself to the KIC. When KIC gets the new user's ID the KIC will generate the new user's public key, private key and certificate. All these secret information will be stored safely in private database that can only be used by the KIC. Directory services, such as LDAP, can be deployed in this infrastructure for users to find out whether the other users are in ID-based domain or not.

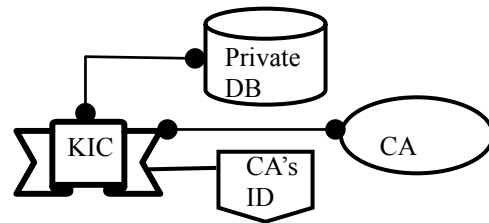


Figure 2 Proxy CA's elements

B. Authentication Phase

In authentication phase, the following steps will be done during the authentication between two different domain users.

(1) Sending an authentication request:

Whenever ID-based domain user A wants to authenticate PKI domain user C, user A queries the KIC in the PCA whether user C's ID has been registered or not. If user C's ID is available, it means that user C is in ID-based domain and therefore ID-based authentication scheme mentioned later in section 3 can authenticate user C in this case.

If this is not the case, it means that user A and user C are in different domains. After the KIC receives this request, the PCA will temporarily represent A to go on further communicate using PKI scheme.

(2) Retrieving the private data:

After the KIC receives authentication request from user A, it will play a role as a proxy of user A. All authentication processes will transform from user A to the KIC for security consideration. In order to authenticate user C in PKI domain, the KIC will retrieve user A's private key and certificate that has been generated in the initialization phase from private database by user A's request.

(3) Querying the certificate from PKI:

After the KIC retrieves user A's private data, the PCA will go on to get user C's private data. The PCA will query user C's certificate from the CA. Moreover, whether user C's certificate is expired or not will also be

checked within this procedure.

(4) Authenticating the PKI domain user:

On behalf of A, the KIC in the PCA has to authenticate user C. Nowadays PKI authentication protocols can be deployed. All these authentication protocols perform mutual authentication between two users who have already got their own certificates from the CA. After authentication is confirmed, a new session key shared by the KIC and user C can be derived at the same time. This shared session key must be stored securely in the PCA.

(5) Authenticating the ID-based domain user:

Since the KIC now authenticates PKI domain user C and retrieves a session key shared with user C, the KIC now can securely communicate with user C. The KIC now needs to authenticate ID-based domain user A using ID-based authentication protocols. A shared session key for the KIC and user A can also be derived. This key must be kept securely in the PCA. Mutual authentication can be achieved after the above-mentioned five steps.

Likewise, when PKI domain user C wants to authenticate ID-based domain user A, user C sends an authentication request to the PCA. Since user A's certificate cannot be found in the CA's directory, the CA uses the ID-based authentication scheme to authenticate the KIC using the CA's ID that has been registered in the initialization phase. Then, the CA temporarily acts as the delegate of user C.

3 ID-based Authentication and Key Exchange

In this section, we detail the design of the ID-based authentication and key exchange protocol based on the architecture presented in previous section. Conditions that our ID-based authentication and key exchange protocol will involve when:

- A. both users are in the same ID-based domain,
- B. an ID-based domain user wants to authenticate users of the PKI domain, and
- C. a user of the PKI domain wants to authenticate users of the ID-based domain.

In first condition, it's quite straightforward using ID-based authentication and key exchange protocol. In second condition, when ID-based domain user wants to authenticate PKI domain user, he has to authenticate the KIC in the PCA first in order to send an authentication request. Next, the KIC has to authenticate the CA in order to get PKI domain user's certificate securely. In each authentication step, one common session key will be generated. Therefore, secure communication can be done using these two session keys.

In third condition, a PKI domain user sends an authentication request to the PCA. The PCA now plays the role of proxy to continue this authentication process. The PCA then authenticates the KIC to get ID-based domain user's certificate. PKI authentication protocols can be deployed now to authenticate ID-based domain user and PKI domain user using their own certificate. Next, the

KIC in PCA needs to authenticate ID-based domain user using ID-based domain authentication protocol and get a shared session key for secure communication. Therefore, authentication and secure communication can be achieved.

The proposed ID-based authentication protocol needs no trusted third party to be participated in authentication process. Only involved identities are needed during authentication process. It owns the property of minimal messages that are needed during the ID-based authentication process. Meanwhile, better security can also be achieved.

The ID-based authentication protocol in this section consists of three phases: registration phase, authenticated key exchange phase and subsequent authentication phase. The registration phase is completed at the key information center to set up the system, and the authenticated key exchange phase is executed between the two communication parties to achieve mutual authentication and exchange the common session key. Finally a subsequent authentication phase is used for subsequent authentication communications.

A. Registration Phase

In this phase, the KIC is responsible neither for mutual authentication nor for the generation of common keys. The KIC is to simply generate public and secret information for newly registered users. When the secure network system is setting up, the key information center will execute the following steps:

Choose two large prime numbers p_1 and p_2 , and let $n = p_1 \cdot p_2$.

- 1) Obtain the KIC's private key P from the following computation, which is only known by the KIC in PCA.

$$3 \cdot P \pmod{\varphi(p)} = \square \tag{1}$$

$$\text{Where } \varphi(p) = (p_1 - 1) \cdot (p_2 - 1)$$

- 2) Find an integer G , which is a primitive element in both $GF(p_1)$ and $GF(p_2)$. We use G as the KIC's public information.
- 3) Let ID_i denote the identity of ID-based domain user i . ID_i could be composed of clear-text form such as name, address, ..., and so on. In our protocol, if two users are in different domains, the ID-based domain user has to authenticate the CA first using the ID of the CA that has been registered in initialization phase. The authentication processes between ID-domain user and CA is done within PCA, and the KIC is viewed as a proxy on behalf of ID-domain user. The CA also has to register itself to KIC using its own identity. Therefore, the ID-domain user, the KIC and the CA in PCA can authenticate one another using the following ID-based authentication and key exchange protocols.

- 4) Choose a one-way hash function $h(x)$ to compute the extended identity (EID_i) of ID-based domain user i as follows

$$EID_i \equiv h(ID_i) \pmod{2^n} \quad (2)$$

- 5) After computing ID-based domain user i 's extended identity EID_i, we calculate the ID-based domain user i 's secret information S_i by the following equation.

$$S_i \equiv EID_i^d \pmod{n} \quad (3)$$

From the relations above, the following equation would be obtained.

$$EID_i \equiv S_i^3 \pmod{n} \quad (4)$$

- 6) Send back $(n, G, h(x), S_i)$ to ID-based domain user i over a secure channel. Upon receipt of the message, user i must keep S_i secret. The ID-based domain user i 's public information is $(n, G, h(x))$.

Once the secure network system is set up, the key information center in the PCA is not needed except when new ID-based domain users join. The center's secret information d must be stored secretly for subsequent use. However, the two integers p and q will be no longer used and should be thrown away secretly. When a new user requests to join, he sends the center his ID. Upon receipt of the user ID, the center repeats steps 5–7. This phase is done in PCA's initialization phase while the PCA sets up in proposed infrastructure.

After the registration phase is done, the KIC in the PCA will automatically generate an ID-based domain user certificate that is to be stored in private database. Traditional CA creates a certificate by signing a collection of information about the entity. This information includes the public key and distinguished name of the entity and may include an optional unique identifier that holds additional information about the entity. The certificate issued by tradition CA contains the following information:

V	The version number of the certificate
SN	The serial number
AI	Identifies the signature algorithm used to sign the certificate
CA	The distinguished name of the issuing CA
UCA	A unique identifier for the issuing CA (optional)
A	The distinguished name of the subject identifier by the certificate
UA	A unique identifier for the subject (optional)
Ap	The public key of subject A.
T ^A	The validity period of the certificate described by a start date and an end data for which the certificate is valid

However, in comparison with tradition CA, the certificates issued by the PCA are quite different from those issued tradition CA. More information needs to be stored in certificates. The certificate issued by the PCA contains the following information:

D	The unique name of the ID-based domain
V	The version number of the certificate
SN	The serial number
AI	Identifies the signature algorithm used to sign the certificate
PCA	The distinguished name of the issuing PCA
UPCA	A unique identifier for the issuing PCA (optional)
A	The distinguished name of the subject identifier by the certificate
UA	A unique identifier for the subject (optional)
Ap	The public key for ID-based domain user A
T ^A	The validity period of the certificate described by a start date and an end data for which the certificate is valid

The symbol "D" represents the unique name of the ID-base domain. The reason the domain name is used is for distinguishing two users have the same name but come from different ID-based domains. Another symbol "Ap" represents the public key of the user. The public key contains $(n, g, f(x))$ and can be distributed with the certificate.

B. Authenticated Key Exchange Phase

Before secure communication starts, each identity needs to authentication another's identity. In the PCA infrastructure, two identities in ID-base domain still need to authenticate each other. Moreover, when two identities are in different domains, it still needs PCA's help to authenticate one another. The proposed authenticated key agreement protocol needs three messages to complete the mutual authentication and key agreement when two users are both in ID-based domain. Upon receipt of the first message from ID-based domain user i , the CA generates a session key, uses it to encrypt a nonce R_4 and send back user i . The user i tries to generate another session key, uses it to decrypt the nonce, and verifies the identity of the CA. If the verification succeeds, he believes that the message is sent by the CA, and they are using the same session key to communicate with each other. In the next step, user i encrypts the other nonce R_2 and sends it back to the CA. The CA then decrypts the nonce and uses it to verify the identity of user i .

Four authenticated key exchange conditions may be happened in the PCA. First, ID-based domain user wants to authenticate the same ID-based domain user. Second, ID-based domain user wants to authenticate PKI domain user. Third, PKI domain user wants to authenticate

ID-based domain user. Fourth, PKI domain user wants to authenticate the same PKI domain user. In first condition and fourth condition, since both users are in the same domain, many authentication and key exchange protocols have been proposed. In this paper, we propose PCA infrastructure that can provide a solution for rest conditions.

In first condition, the execution steps for mutual authentication and key agreement for a session are listed as follows.

- 1) If ID-based user i in domain I ($I - ID_i$) wishes to communicate the CA in PKI domain P ($P - ID_{CA}$), he generates two random numbers R_1 and R_2 , and calculates the following two integers:

$$T_1 = G^{3 \cdot R_1} \pmod{n}. \quad (5)$$

$$T_2 \equiv S_i \cdot R_2 \cdot G^{2 \cdot R_1} \pmod{n}. \quad (6)$$

where R_2 is used for challenge.

- 2) User i sends these three integers T_1 , T_2 , and R_2 together with ID_i to the CA in PKI domain.
- 3) Upon receipt the message, the CA in PKI domain checks whether the following equation holds:

$$EID_i \cdot R_2^3 = T_2^3 / T_1^2 \quad (7)$$

- 4) If it is true, the CA generates two random numbers R_3 and R_4 , and calculates the following two integers:

$$Q_1 = G^{3 \cdot R_3} \pmod{n} \quad (8)$$

$$Q_2 = S_{CA} \cdot R_2 \cdot G^{2 \cdot R_3} \pmod{n} \quad (9)$$

- 5) The CA calculates a session key K_{CA-i} from T_1 , and uses the key to encrypt the integer R_2 :

$$K_{CA-i} \equiv T_1^{R_2} \equiv G^{3 \cdot R_1 \cdot R_2} \pmod{n} \quad (10)$$

$$Z_{CA} = \{R_2\}K_{CA-i} \quad (11)$$

This session key must be kept securely in PCA.

- 6) The CA sends these four integer Q_1 , Q_2 , R_2 , and Z_{CA} along with ID_{CA} to ID-based domain user i .
- 7) In the same way, upon receipt of the message, calculates $EID_{CA} = h(ID_{CA})$ and checks whether the following equation holds:

$$EID_{CA} \cdot R_4^3 = Q_2^3 / Q_1^2 \quad (12)$$

- 8) If it is true, user i calculates the session key K_{i-CA} from Q_1 :

$$K_{i-CA} \equiv Q_1^{R_2} \equiv G^{3 \cdot R_1 \cdot R_2} \pmod{n} \quad (13)$$

- 9) ID-based domain user i try to decrypt R_2 by K_{i-CA} and verify if it's the same as he sent to the CA. If it is true, user i believes the message is sent by the CA, and they are using the same session key.
- 10) User i uses the session key K_{i-CA} to encrypt R_4 and sends it to the CA:

$$Z_i = \{R_4\}K_{i-CA} \quad (14)$$

Upon receipt the message, the CA tries to decrypt it by his session key Key_{CA-i} . If he gets the correct R_4 , the CA believes he is communicating with ID-based domain user i , and agrees to use same session key to encrypt the future communicating messages. The third message (Z_{CA}) can be sent with normal packets to reduce traffic overhead. This phase is done when ID-based domain user sends an authentication request and finds that the user he wants to authenticate is also in the same ID-based domain. In this phase, one common session key is derived for secret communication.

In second condition, when ID-based domain user wants to authenticate PKI domain user, it needs to do the following steps:

- 1) ID-based domain user i (ID_i) sends an authentication request to the PCA for authenticating PKI domain user j (PKI_j).
- 2) The KIC in PCA plays a role of proxy on behalf of ID_i to authenticate PKI_j . It uses ID_i and CA's ID (ID_{CA}), being registered in initialization phase, to authenticate each other. A session key, KEY_{KIC-CA} , which is shared by the KIC and the CA, will be generated in this step.
- 3) The KIC retrieves ID-based domain user i 's certificate from its private database.
- 4) The KIC uses the session key, KEY_{KIC-CA} , to get PKI domain user j 's certificate.
- 5) The PKI authentication protocol can be used to authenticate each other using ID-based domain user i 's certificate and PKI domain user j 's certificate, where the KIC plays a role of proxy on behalf of ID-based domain user i . Another session key ($KEY_{KIC-PKI_j}$) will be derived now. This session key now is shared by the KIC and the PKI domain user j .
- 6) The KIC turns to authenticate ID-based domain user i using proposed ID-based authentication protocol. The KIC uses the CA's ID and user i 's ID to proceed with the authentication process. A session key (KEY_{KIC-ID_i}) shared by the KIC and ID-based domain user i will be generated.

By using the above steps, mutual authentication and secure communication can be achieved in second

authentication condition between ID-based domain user and PKI domain user.

In third condition, when PKI domain user wants to authenticate ID-based domain user, it needs to do the following steps:

- 1) PKI domain user j (PKI_j) sends an authentication request to the PCA for authenticating ID-based domain user i (ID_i).
- 2) The KIC in PCA will get ID-based domain user i 's certificate from its private database.
- 3) Since both of them have their own certificate, the PKI authentication can be used to authenticate each other. The authentication procedure relies mainly on the PKI authentication protocols.

By using the proposed protocol and with the help of the PCA, the authentication and key exchange protocols between two different domains can then transform to only in one domain. In first and second condition, it turns to using ID-based authentication and key exchange protocols. In third and fourth condition, it changes to use PKI authentication and key exchange protocols. Each mechanism then can easily authenticate one another no matter what domain they are in.

C. Subsequent authentication phase

This phase represents the subsequent authentication steps for ID-based domain users to authenticate the others. In the PCA, the same steps must also be performed to authenticate the KIC and the CA so that the certificates from PKI domain can be obtained securely.

After the initial authentication, the nonce variables R_2 and R_4 can be used for subsequent authentication. The new nonce variables R'_2 and R'_4 can be derived from $h(R_2)$ and $h(R_4)$, respectively, where $h(x)$ is the same one-way function used in initial phase. Since both parties know the value of R_2 and R_4 from a previous session, the delivery of R_2 and R_4 are not required, which means only two messages is needed in the subsequent authentication phase. A new session key K'_{i-CA} will be generated for each subsequent authentication, and only used in a session. Since the session keys will not be reused, replay attacks can be prevented. Both the communication parties can determine the freshness of the reply by checking the value of R_2 and R_4 . At the end of subsequent authentication, the client encrypts the succeeding data message in this session with the new session key. The execution steps for subsequent authentication of a session are listed as follows.

- 1) If ID-based user i in domain I ($I-ID_i$) wishes to communicate with the CA in PKI domain K ($P-ID_{CA}$) again, he generates one random numbers R'_1 , and calculates the following two integers:

$$T'_1 = G^{3 \cdot R'_1} \pmod{n}. \quad (15)$$

$$T'_2 = S_i \cdot R_2 \cdot G^{2 \cdot R'_1} \pmod{n}. \quad (16)$$

- 2) ID-based user i sends these two integers together with ID_i to the CA in PKI domain.

- 3) Upon receipt the message, the CA generates another random numbers R'_3 , and calculates the following two integers:

$$Q'_1 = G^{3 \cdot R'_3} \pmod{n} \quad (17)$$

$$Q'_2 = S_j \cdot h(R_2) \cdot G^{2 \cdot R'_3} \pmod{n} \quad (18)$$

- 4) The CA in PKI domain calculates a session key K'_{CA-i} from Q'_1 :

$$K'_{CA-i} = (T'_1)^{R'_3} = G^{3R'_1 \cdot R'_3} \pmod{n} \quad (19)$$

- 5) The CA checks whether the following equation holds:

$$EID_i \cdot (h(R_2))^3 = (T'_2)^3 / (T'_1)^2 \quad (20)$$

- 6) If it's true, the CA in PKI domain believes the message was sent from ID-based domain user i , he will send these two integers Q'_1 and Q'_2 , along with ID_{CA} to user i .

- 7) Upon receipt of the message, user i calculates $EID_{CA} = h(ID_{CA})$ and checks whether the following equation holds:

$$EID_j \cdot (h(R_4))^3 = (Q'_2)^3 / (Q'_1)^2 \quad (21)$$

- 8) If the equation holds, user i believes the message is sent by the CA. Now user i calculates the session key K'_{i-CA} from Q'_1 :

$$K'_{i-CA} = (Q'_1)^{R'_1} = G^{3 \cdot R'_1 \cdot R'_1} \pmod{n} \quad (22)$$

During ID-based authentication and key exchange procedures, the proposed subsequent authentication protocol takes three messages to finish the authentication procedure. Two messages are needed to finish the key exchange procedure.

In the proposed protocol, each domain's user has to authenticate and exchange keys with the PCA. The protocol has the property of using minimal messages to provide better security with totally six messages needed for authentication and four messages for key exchange.

4. Security Analysis

After the ID-based authentication and key exchange procedures finish in the PCA and regardless of the domain that the user registers with, the user can authenticate the other users using a session key generated by the KIC in the PCA. Then, both users can use the session key to initiate a secure session with each other. In the following, we discuss the security analysis of the proposed protocol. Our protocol provides session key exchange and the

authenticity of communicating parties to guarantee the privacy and security. The security of our protocol relies on the difficulty of computing the discrete logarithm problem [17], which does not have the conspiracy problem existing in the Tsujui's scheme [17]. If a forger wants to masquerade user i and tries to communicate with others, he must find two integers x and y satisfying the following equation:

$$T_2^3 = EID_i \cdot N_i^3 \cdot T_1^2 \quad (23)$$

The use of low public exponents in this equation does not lower the difficulty to crack (T_2, T_1) . Although the forger can get a pair of integers (T_2^3, T_1^3) that makes the equation hold, the pair (T_2, T_1) is unattainable because computing (T_2, T_1) pair from (T_2^3, T_1^3) is a discrete logarithm problem.

In our protocol, a small integer 3 is used as the public exponent in equation (1). The selection of small integer is for the purpose of reducing computation overhead. For a general public exponent e , our protocol still works fine. The use of a low public exponent does not lower the difficulty to crack the user secret information S_i , because the computation of S_i in equation (4) is a discrete logarithm problem [5]. Though some low exponent attacks [2][5] are proposed, both of them do not work in our protocol. Hastad [5] proposed an attack on using RSA with low exponents in a public key network [5]. To illustrate this attack, suppose that a message m is broadcasted to three parties in which the public exponents are $e_1 = e_2 = e_3 = 3$, and in which the modulus are n_1, n_2 , and n_3 . The encrypted messages are $m^3 \bmod n_1, m^3 \bmod n_2$ and $m^3 \bmod n_3$.

Using the Chinese remainder theorem, one can find $m^3 \bmod n_1 \cdot n_2 \cdot n_3$. However, $m^3 < n_1 \cdot n_2 \cdot n_3$ because $m < n_1 \cdot n_2 \cdot n_3$. Therefore, m^3 is not affected by being reduced modulo $n_1 \cdot n_2 \cdot n_3$. This attack will not succeed in our protocol, because the same modulus n is used for all parties. This attack will not succeed, since our protocol uses the same modulus n for all parties. In 1996, Coppersmith and et al [1] presented another low exponent attack, in which encrypted messages may be recovered under RSA with a public exponent of 3, if a cracker can get α β m and m_2 satisfy $m_2 = \alpha \bullet m_1 + \beta$. This attack will not work in our protocol, since an outsider has no idea of the relation of users' secret information S_i to crack equation (4).

Conclusions

In this paper, we have presented an ID-based proxy authentication protocol that can be interoperable with PKI. The proposed protocol leverages the management mechanisms of PKI while inheriting the nature of traditional ID-based protocols. ID-based authentication

protocol needs no trusted third party to be joined during authentication processes. However, PKI authentication needs the CA to authenticate users and to check the validity of a certificate. Authentication between two entities of different domains is made possible with the help of a proxy CA presented in this paper. ID-based authentication and PKI authentication protocol can both work alone well in each domain and can be worked together with the aid of the PCA.

When performing the authentication between ID-based and PKI domains, two session keys will be derived in each domain. If secure communication is required, the PCA must also be the gateway that bridges the traffic. As a result, the PCA will be a bottleneck. The enhancement of the proposed protocol that eliminates the need of two session keys will be left as the further work of this paper.

References:

- [1] Levi, A., Caglayan, M. U., "An Efficient, Dynamic and Trust Preserving Public Key Infrastructure", *IEEE Symposium on Security and Privacy*, pp. 203-214, 2000.
- [2] D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter, "Low-exponent RSA with Related Messages," *Advances in Cryptology-Eurocrypt 96*, pp. 1-9, 1996.
- [3] W. Diffie, "Authentication and Authenticated Key Exchanges," *Design, Codes, and Cryptography*, pp.107-125, Vol. 2, 1992.
- [4] W. Diffie and M. E. Hellman, "New Directions in Cryptography," *IEEE Transactions on Information Theory*, Vol. 22, No. 6, pp. 644-654, 1976.
- [5] Hastad, "On using RSA with low exponent in a public key network," in *Lecture Notes in Computer Science: Advances in Cryptology CRYPTO'85 Proc.*, pp. 403-408.
- [6] Ray Hunt, "PKI and Digital Certification Infrastructure", *9th IEEE International Conf. On Networks*, 2001.
- [7] Kai Hwang, "Wireless PKI and Distributed IDS for Securing Intranets and M-Commerce", *IEEE 3rd International Conf. On Parallel and Distributed Computing, Applications and Technologies*, 2002.
- [8] R. E. Lennon, "Cryptography Architecture for Information Security," *IBM Systems Journal*, Vol. 17, No. 2, pp. 138-150, 1978J.
- [9] E. Okamoto and K. Tanaka, "Identity-based information security management system for personal computer networks," *IEEE Journal on Selected Areas In Communications*, vol. 7, pp. 290-294, Feb. 1989.
- [10] Radia Perlman, "An Overview of PKI Trust Models", *IEEE Network*, Nov. 1999.
- [11] R. L. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signature and Public-key Cryptosystem," *Communications of the ACM*, Vol. 21, No. 2, pp. 120-126, 1978.

- [12] A. Shamir, "Identity-based cryptosystems and signature schemes," *Proceeding of Crypto-84*, Santa Barbara, CA, 1984, pp. 47-53.
- [13] Shih-Pyng Shieh, Jia-Ning Luo, "An ID-Based Authenticated Key Agreement Protocol", *ACM International Conf. On Info. Security*, 2002.
- [14] Shih-Pyng Shieh, Wen-Her Yang, and Hun-Min Sun, "An Authentication Protocol Without Trusted Third Party," *IEEE Communication Letters*, pp.87-89, Vol. 1, No. 3, May 1997.
- [15] Hunt, R., "PKI and Digital Certification Infrastructure", *Ninth IEEE International Conf. On Networks*, 20001.
- [16] Chokhani, S., "Towards a National Public Key Infrastructure", *IEEE Communications Magazine*, vol. 32, no. 9, pp.70-74, Sep. 1994.
- [17] Tsujii, T. Itho, and K. Kurosawa, "ID-based cryptosystem using discrete logarithm problem," *Electronic Letter*, vol. 23, pp. 1318-1320, Nov. 1987.