Threshold PKC를 이용한 패스워드 기반 키분배 방식

A Password-Authenticated Key Distribution Method Using Threshold PKC

이영숙*, 이영교**, 원동호**
성균관대학교 정보통신대학원*, 성균관대학교 정보통신공학부**
Young-Sook Lee(ysl472@yahoo.co.kr)*
Young-Gyo Lee(leeyounggyo@hanmail.net)**, Dong-Ho Won(dhwon@ece.skku.ac.kr)**

요약

본 논문에서는 Threshold PKC를 이용하여 패스워드 사전공격에 안전한 사용자인증 및 키분배 방법을 제안한다. 신뢰기관의 개인키를 (t,n) 비밀분산을 이용하여 n개의 서버에 분산 저장시키고 서버가 사용자를 인증할 때 t개 이상의 서버가 협력하여 사용자의 패스워드 검증자를 복원한다.

□ 중심어 : | 비밀분산 | 사용자 인증 |

Abstract

In this paper we present user authentication and key distribution using threshold PKC(Public Key Cryptosystem), which is secure against the dictionary attack. The n servers hold a t-out-of-n sharing of the dealer's secret key. When the server authenticate a user, at least t of them cooperate they can reconstruct password verifier.

□ keyword : | Secret Sharing | User Authentication |

I. 서 론

비밀을 여러 개의 조각으로 나눈 후 분산 시켜 저장하고 있다가 비밀복원이 필요할 때 threshold(임계값)를 적용하여 비밀을 복원할 수 있는 기술을 Threshold Cryptosystem이라 하는데 이 기술을 공개키 암호 방식의 개인키를 분산 저장하거나 전자서명에 이용한 것이 Threshold PKC(Threshold Public key Cryptosystem)이다.

공개키 암호방식은 평문을 암호화시키는 암호화키인 공개키는 공개목록에 공개하고 암호문을 평문으로 복호하거나 전자서명을 할 수 있는 개인키는 비밀리에 보관한다.

기존의 공개키 암호방식은 개인키를 개인이 보관하거나 한 개의 저장매체에 보관하는데 비해 Threshold PKC는 개인키를 보관하는 여러 개의 서버를 두고 그 개인키를 조각내어 분산 저장시킨 후 필요할 때 개인키를 복원하거나 또는 암호문을 여러 개의 분산 서버에 보내 부분 복호를 수행하게 한 후 그것들을 모아서 평문으로 복호할 수 있다.

접수번호 : #040604-001
접수일자 : 2004년 6월 4일
심사완료일 : 2004년 7월 7일
교신저자 : 이영숙, e-mail : ysl472@yahoo.co.kr

제안하는 방식은 다중 서버들과 비밀 통신을 하는 환경에서 사용자는 미리 신뢰기관에 사용자등록을 하고 사전에 등록하지 않은 서버들과 비밀 통신을 하려고 할 때 신뢰기관을 통하지 않고 직접 통신하고자 하는 서버에 사용자 인증을 의뢰한다. 그러면 사용자와 비밀통신을 하려는 서버는 신뢰기관의 개인키를 분산 저장하고 있는 서버들과 서로 협력하여 사용자를 인증한다. 어떤 서버가 사용자 인증을 해야 할 때 다른 서버들과 협력하여 신뢰기관의 개인키를 노출시키지 않고 사용자 인증 및 키분배 하는 방법을 제안한다.

2장에서는 Threshold PKC를 소개하고 3장에서는 제안하는 방식을 4장에서는 결론을 맺는다.

II. Threshold PKC(Public Key Cryptosystem)

1. Shamir의 (t,n) 비밀분산(secret sharing)

Shamir가 제안한 비밀분산 방식은 비밀을 S라 할 때 신뢰할 수 있는 딜러가 임의의 t차 다항식 $f(z)$을 생성하여 비밀분산에 참가하는 n명의 참가자에게 비밀조각 S_i을 분배한 후 비밀복원을 할 때 참가자 n명 전원이 프로토콜에 참여하는 방식이 아니고 threshold인 t이상이 모여서 비밀을 복원하는 방법이다[1].

$$f(z) = S + \sum_{j=1}^{t} a_j z^j \text{ (S: 비밀)}$$

$$S_i = f(i) \bmod q \ (q : \text{prime}) \ (i = 1, \ldots, n)$$

이 비밀분산의 단점은 비밀복원에 참여하는 참가자들이 자신들이 가지고 있는 비밀조각이 정당한 것인지 확인할 수 없고 비밀이 한번 복원된 이후에는 참가자들은 부분정보를 재사용 할 수 없다는 단점이 있다. 참가자의 집합이 변하는 경우 비밀정보를 분산하는 딜러(dealer)는 새로운 다항식을 생성하여 기존의 참가자들에게 새로운 부분 분산정보를 재분배해야 하는 문제가 있다.

비밀조각의 정당성을 검증할 수 있는 비밀분산 방식을 VSS(Verifier Secret Sharing)이라 하는데 Pedersen's VSS와 Feldman's VSS가 있다[6, 7].

표 1. VSS의 비교

	Feldman's VSS	Pedersen's VSS
안전성	이산대수 문제	이산대수 문제
비밀 분배	t차 다항식이용	두 개의 t차 다항식이용
검증식	$g^{f(i)} = \prod (b_j)^{i^j} \bmod N$	$D_j = G^{a_j} \cdot H^{r_j} \bmod N$ $D_j = G^{b_j} \cdot H^{c_j} \bmod N$ $G^{k(i)} \cdot H^{g(i)} = \prod D_j^{i^j} \bmod N$
공개 입력	RSA modules N	RSA modules N
비밀 복원	Lagrange보간법이용	Lagrange 보간법이용

2. Catalano가 제안한 Threshold PKC

2.1 개요

공개키 암호방식에서 개인키를 소유하는 single party대신 개인키 조각을 소유하여 부분복호(partial decryption)를 담당하는 n개의 서버를 두고 수신자는 암호문 C를 받았을 때 복호를 담당하는 n개의 서버에 암호문을 전송하여 그들로 하여금 복호를 하게 하여 부분복호된 것들을 모아서 평문을 복호하는 방법을 제안하였다[3].

공개키 암호방식을 E라 하고 공개키 암호방식을 이용한 Threshold Cryptosystem을 T_E라 할 때 T_E는 두개의 프로토콜로 구성한다.

- threshold 키 생성 프로토콜

공개키 암호방식에서 암호화를 수행하는 공개키 EK는 공개하고 개인키 DK를 (t,n) 비밀분산을 이용하여 $DK_1, \cdots, DK_n$으로 조각내어 분산저장 하며 참가자 P_i는 부분정보 DK_i를 비밀리에 저장하는 프로토콜이다.

- threshold 복호화 프로토콜

참가자 P_i가 평문 M에 대한 암호문 $C = E_{EK}(M)$를 공개 입력으로, 부분정보 DK_i는 비밀입력으로 하여 부분복호를 수행하여 평문 M을 공개 출력하는 과정을 담

당하는 프로토콜이다.

위의 프로토콜은 수행되는 동안 개인키 DK에 대한 어떤 정보도 드러내지 않는다는 장점이 있다.

2.2 ElGamal 방식을 이용한 Threshold PKC

이 프로토콜은 암호문을 생성하는 송신자와 암호문을 수신하는 수신자, n개의 프락시(proxy) 서버로 구성되어 있고 수신자와 프락시 서버 사이는 private channel과 broadcast channel을 갖는 통신환경에 있고 동기식 전송을 한다. 암호 방식은 ElGamal 방식을 이용하며 비밀을 분산 저장 시키는 Threshold ElGamal Key Generation(TEG-Key-Gen.)과 복호를 담당하는 Threshold ElGamal Decryption(TEG-Decrypt.) 두 부분으로 구성되어 있다[4]. TEG-Key-Gen. 은 두 소수의 곱인 N을 파라미터로 사용하고 개인키에 해당하는 비밀정보인 x를 임의로 선택해서 분산저장 시키고 각 참가자들은 $\phi(N)$을 알 수 없다. 암호문을 복호화하는 TEG-Decryp. 프로토콜은 비밀정보의 additive share를 이용해서 암호문에 지수승하여 부분복호를 수행하는데 이때 비밀정보는 각 참가자들의 additive share들의 합이고 부분 복호된 정보들의 곱이 정당한 복호가 된다.

이 논문에서 송신자는 자신의 개인키로 메시지를 서명하여 수신자에게 보냈을 때 수신자는 송신자의 공개키로 메시지를 확인하지 않고 자신의 공개키로 암호화시킨 후 프락시 서버로 전송한다. 수신자는 n개의 프락시 서버에 자신의 개인키를 (t, n) 비밀분산을 이용하여 분산 저장 시킨다. 이때 프락시 서버들은 수신자에게서 전송된 암호문을 저장된 분산정보를 이용하여 부분복호 하여 그 조각들을 수신자에게 보낸다. 그러면 수신자는 프락시 서버로부터 받은 정보들을 모아서 메시지를 획득하는 방법을 제안 하였다[3].

▫ ElGamal 방식을 이용한 threshold 키 생성 프로토콜(TEG-Key-Gen.)

입력 : N(두 소수의 곱), $g(Z_N^*$상의 최대위수를 갖는 원소)

$L = n!$ (n은 참가자의수)

1) 각 참가자 P_i는 임의의 $x_i \in_R [-N^2 ... N^2]$와 t개의 임의의 상수 $a_{i,t}, ..., a_{i,1} \in_R [-L^2N^3 ... L^2N^3]$을 선택하여 임의의 t차 다항식 $f_i(z) = Lx_i + a_{i,1}z + \cdots + a_{i,t}z^t$를 생성한다.

2) 참가자 P_i는 $\alpha_{i0} = g^{x_i}, \alpha_{ik} = g^{a_{i,k}} \bmod N$을 계산하여 공개한다.

3) 참가자 j에게서 받은 정보를 x_{ji}라 할 때 $x_{ji} = f_j(i)$라 놓고 (1)식으로 검증하여 참가자 P_j가 정당한 정보를 제공하는지 확인한다.

$$g^{x_{ji}} = \alpha_{j0}{}^{L} \prod \alpha_{jk}{}^{i^k} \bmod N \quad \cdots (k = 1, ..., t) \tag{1}$$

4) 만일 참가자 P_j가 부당한 정보를 제공했다면 단계 1)의 VSS를 사용하여 x_i를 x_j로 새로이 정하고 $\alpha_{i0} = g^{x_j} \bmod N$으로 바꾸어 놓는다.

5) 공개키 $Y = \prod \alpha_{i0} \bmod N$을 공개 출력으로, 참가자의 정당한 부분정보 x_i는 비밀 정보로 저장한다.

▫ ElGamal 방식을 이용한 threshold 복호화 프로토콜(TEG-Decryt.)

모든 참가자의 입력 : N(두 소수의 곱), $g(Z_N^*$상의 최대위수를 갖는 원소), TEG-Key-Gen.의 공개출력, ElGamal 암호방식으로 평문M을 암호화한 암호문 (C_1, C_2) : 암호문 C_1는 송신자의 공개키, $C_1 = g^K \bmod N$, 암호문 C_2는 평문 M을 ElGamal 방식으로 암호화한 암호문

$C_2 = Y^K \cdot M \bmod N \ (K \in_R Z_N)$

Y는 수신자의 공개키,

$Y = g^x \bmod N,\ x \in_R Z_N$

비밀입력 : 참가자 P_i의 정당한 부분정보

1) 참가자 P_i 의 정당한 부분정보를 이용하여 암호문의 일부인 C_1을 부분 복호하여 공개한다. 즉, 부분복호 $A_i = C_1^{x_i} \bmod N$을 공개한다.
이때 부분복호는 $DLog_{C_1} A_i = DLog_g \alpha_{i0}$를 만족해야 한다.
2) 만일 단계1)의 검증식을 만족하지 않으면 새로운 x_i값을 계산하여 $A_i = C_1^{x_i} \bmod N$을 수행한다. $(i = 1, \cdots, t)$
3) 단계2)의 부분 복호된 것을 모아서 평문을 출력한다.

$$M = \frac{C_2}{\Pi A_i} \bmod N \quad (i = 1, \cdots, t)$$

복호를 담당하는 참가자들은 딜러의 개인키를 모르는 상태로 부분복호를 수행하여 평문을 알 수 있다.

III. 제안하는 방식

1. 사용자 등록과정

사용자는 Z_N에 속하는 임의의 a를 선택하여 공개키 g^a(g는 Z_N^*상의 최대위수를 갖는 원소)를 생성하고, 자신의 패스워드를 해쉬함수에 입력한 값을 패스워드 검증자를 g_π라 할 때, $g_\pi = h(pwd)$를 ElGamal 암호 방식을 이용하여 딜러(신뢰기관)에게 등록한다.

사용자	g^a, g^x 공개	딜러
$a \in_R Z_N$ $C_1 = g^a \bmod N$ $C_2 = g_\pi \cdot (Y)^a \bmod N$	(ID,(C_1,C_2)) →	$x \in_R Z_N$ $Y = g^x \bmod N$ $\frac{C_2}{(C_1)^x} \bmod N$ $= g_\pi$

그림 1. 사용자 등록과정

2. 서버의 등록과정

딜러는 사용자 등록과정에서 확인된 패스워드 검증자를 분산된 서버에 (ID, g_π) 형태로 저장한다.

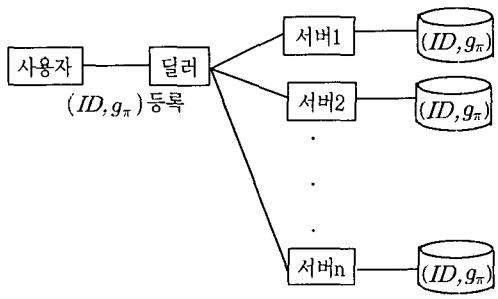

그림 2. 서버의 사용자(ID, g_π) 등록과정

3. 사용자 인증

사용자는 비밀통신을 하려는 서버i에게 ElGamal 암호방식을 사용하여 자신의 패스워드 검증자 g_π를 보낸다. 서버 i는 딜러의 개인키를 분산 저장하고 있는 다른 서버들에게 사용자에게서 전송되어온 암호문의 일부분(C_1)을 보내어 부분복호를 수행하게 하고 그것들을 모아서 사용자의 패스워드(g_π)를 확인하여 사용자를 인증한다. 사용자와 서버 사이는 private channel과 broadcast channel을 갖는 통신환경에 있고 동기식 전송을 한다. 모든 서버가 동시에 공격받지 않는 것을 전제로 하며 안전성을 고려해서 $n > 3t$ (n은 분산 서버의 수이고 t는 threshold값)로 한다[11].

3.1 서버의 사용자 인증과정

딜러의 개인키를 n개의 서버에 분산 저장한 후 서버

i는 사용자에게서 받은 암호문의 일부를 다른 서버들에게 전송하면 다른 서버들은 부분복호를 수행하여 공개하고 서버i는 그것들을 모아 딜러의 개인키가 노출되지 않은 상태에서 암호문을 복호하여 사용자의 패스워드 검증자 g_π를 확인하여 사용자를 인증한다. 그 과정은 다음과 같다.

▫ ElGamal 방식을 이용한 threshold 키 생성 프로토콜(TEG-Key-Gen.)

모든 서버의 입력 : 두 소수의 곱인 합성수 N, Z_N^*상의 최대위수를 가지는 원소 g, $L=n!$

1) 각 서버 i는 임의의 수 $x_i \in_R [-N^2 \cdots N^2]$와 t개의 임의의 상수 $a_{i,t}, \cdots, a_{i,1} \in_R [-L^2N^3 \cdots L^2N^3]$을 선택하여 임의의 t차 다항식 $f_i(z) = Lx_i + a_{i,1}z + \cdots + a_{i,t}z^t$를 생성한다. 각각의 서버들은 다항식 $f_i(z)$를 사용하여 비밀을 분산 저장시키는 무조건적으로 안전한 VSS을 수행한다.
2) 서버 i는 $\alpha_{i0} = g^{x_i}, \alpha_{ik} = g^{a_{i,k}} \bmod N$을 계산하여 공개한다.

사용자		서버 i
$r \in_R Z_N$ $C_1 = g^r$ $C_2 = g_\pi \cdot (g^x)^r \bmod N$	(C_1, C_2) 전송 →	C_1을 다른 서버들에게 전송

그림 3. 사용자와 서버간의 초기 등록

3) 서버 j에게서 받은 정보를 x_{ji}라 할 때 $x_{ji} = f_j(i)$라 놓고 (2)식으로 검증하여 서버i가 정당한 부분정보를 제공하는지 확인한다.

$$g^{x_{ji}} = \alpha_{j0}{}^{L} \prod \alpha_{jk}{}^{i^k} \bmod N \quad \cdots (k = 1, \cdots, t) \tag{2}$$

4) 만일 서버i 가 부당한 정보를 제공했다면 단계1)의 VSS를 사용하여 x_i를 x_j로 새로이 정하고 $\alpha_{i0} = g^{x_j} \bmod N$으로 바꾸어 놓는다.
5) 공개출력 : 딜러의 공개키 $Y(Y = g^x \bmod N)$
$Y = \prod \alpha_{i0} \bmod N \ (i = 1, \cdots, t)$
비밀출력 : 참가자 P_i의 정당한 부분정보

▫ ElGamal방식을 이용한 threshold 복호화 프로토콜 (TEG-Decryt.)

모든 서버의 입력 : 두 소수의 곱인 합성수 N, Z_N^*상의 최대위수를 갖는 원소 g TEG-Key-Gen의 공개 출력, 서버i로부터 받은 암호문 $C_1(C_1 = g^r)$
각 서버의 비밀입력 : TEG-Key-Gen.의 비밀출력 (참가자 P_i의 정당한 부분정보)

1) 서버 P_i의 부분복호인 $A_i = C_1^{x_i} \bmod N$을 계산하여 공개한다. 이때 부분복호는 검증식 $DLog_{C_1} A_i = DLog_g \alpha_{i0}$를 만족해야 한다.
$(i = 1, \cdots, t)$
2) 만일 단계1)의 검증식을 만족하지 않으면 새로운 x_i값을 계산하여 $A_i = C_1^{x_i} \bmod N$을 수행한다.
3) 서버 i는 부분복호된 것들을 모아서 사용자의 패스워드 검증자 g_π를 확인한다.

$$g_\pi = \frac{C_2}{\prod A_i} \bmod N \quad (C_2 = g_\pi \cdot (g^x)^r \bmod N)$$

부분복호를 담당하는 서버들은 딜러의 개인키를 모르는 상태로 부분복호를 수행하며 서버 i는 그들을 모아서 완전한 평문으로 복호하여 사용자를 인증한다.

4. 키분배 과정

[그림 4]는 키분배 과정을 나타내며 그 과정은 다음과 같다.

1) 서버i는 서버의 사용자 인증과정에서 확인된 g_π와 사용자에게서 받은 암호문 $C_2(C_2 = g_\pi \cdot (g^x)^r)$를

이용하여 $\mu(\mu = \frac{C_2}{g_\pi} = (g^r)^r)$ 를 형성한다.

2) 서버 i 는 임의의 s를 선택하여 이산대수문제를 이용하여 $\sigma(\sigma = (g^r)^s)$를 만든 후 사용자에게 전송한다.

3) 사용자는 서버i에게 보낼 때 사용한 임의의 수 r을 전송받은 σ에 지수승 하여 세션키(session key) K를 만든다.

$$K = \sigma^r$$
$$= g^{rrs}$$

4) 서버i는 1)에서 생성한 μ에 임의의 수 s를 지수승하여 세션키 K를 만든다.

$$K = \mu^s$$
$$= g^{rrs}$$

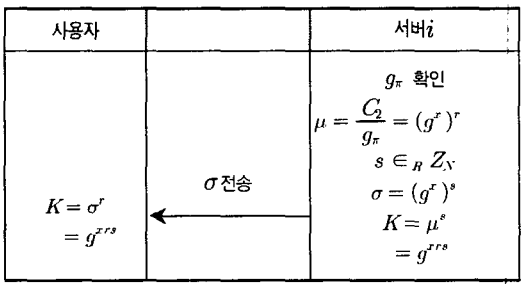

그림 4. 키분배 과정

이때 생성된 세션키는 매번 통신할 때 사용자와 서버가 선택하는 임의의 수에 따라서 매번 바뀌게 되므로 key freshness를 제공한다. 생성된 세션키는 키동의 방식을 따르며 묵시적 키인증을 한다.

5. 안전성 분석

패스워드기반 키분배 방식은 서버해킹이나 사전(dictionary)공격에 약한 단점이 있으므로 복수의 서버를 두고 복수개의 서버가 협력해야만 사용자인증이 가능하도록 하였으며 제안한 방식은 공격자들이 모든 서버를 동시에 공격하지 않는다는 전제하에 안전하고 서버의 개수 n과 threshold t사이에는 $n > 3t$일 때 공격으로부터 안전하다[11].

제안 한 방식은 이산대수문제(Discrete Logarithm Problem)와 DHP(Diffie-Hellman Problem)의 안전성에 의존한다[5]. 세션키는 매번 사용자와 서버 간에 임의의 수를 사용하여 생성하므로 key freshness를 제공하고 노출된 세션키에는 패스워드에 대한 정보가 없으므로 노출된 세션키를 추측한 패스워드에 대한 검증값으로 사용할 수 없다. 따라서 "Danning-Sacco attack"에 대해서도 안전하다.

Ⅳ. 결 론

제안하는 방식은 다중 서버를 이용하는 환경에서 믿을 만한 신뢰기관을(딜러) 통하여 사용자의 패스워드를 미리 등록 해 놓고 각 서버들과 비밀통신을 한다. 서버들은 Catalano가 제안한 threshold secret sharing을 이용하여 신뢰기관의 개인키를 조각내어 분산 저장한다. 나중에 사용자가 서버 i와 비밀통신을 하고자 할 때 서버 i에게 자신의 패스워드 검증자를 신뢰기관의 공개키를 이용하여 암호화해서 보내면 서버 i는 다른 서버들에게 암호문의 일부를 보내서 부분 복호를 의뢰한다. 신뢰기관의 개인키를 분산저장하고 있던 다른 서버들은 전송받은 암호문을 부분 복호하여 공개하고 서버i는 부분복호 된 것들을 모아 사용자를 인증하도록 제안했다. 신뢰기관의 개인키를 (t, n) 비밀분산 방법을 이용하여 복원함으로써 서버 단독으로 패스워드 검증자를 복원할 수 없도록 하여 패스워드 검증자가 공격자에게 노출되더라도 t개의 서버가 공격당하지 않는 한 패스워드 추측공격이 불가능하도록 하였다. 특정 서버가 비밀복원을 하는 과정 중 신뢰기관의 개인키가 복원되는 것이 아니라 신뢰기관의 공개키가 복원되므로 신뢰기관의 개인키도 보호되는 장점을 지닌다.

참 고 문 헌

[1] A. Shamir, "How to share a secret," Communications *of the ACM*, 22 : 612-613, 1979.

[2] R. Cramer and V. Shoup, "Signature Based on the strong RSA Assumption," *To appear in the proceedings of the 6th ACM Conference in Compute and Communication Security*, 1999.

[3] Dario Catalano and Rosario Gennaro, "New Efficient and Secure Protocols for Verifiable Signature Sharing and *Other Applications," CRYPTO '98, LNCS, 1462, pp.105-121, 1998.*

[4] *T. ElGamal, "A public-key cryptosystem and a signature* scheme based on discrete logarithms," IEEE Transactions on Information Theory, IT-31(4) : pp.469-472, 1985.

[5] W. Diffie and M. E. "Hellman, New Direction in Cryptography," IEEE Transactions on Information Theory, V. IT-22. No.6, pp. 644-655, 1976.

[6] P. Feldman, "A Practical Scheme for Non -Interactive Verifiable Secret Sharing," FOCS'87, pp.427-437, 1987.

[7] T. Pedersen, "Non-interactive and information -theoretic secure verifiable secret sharing," CRYPTO'91, pp.129-140, LNCS Vol. 576, 110, 1991.

[8] R. Gennaro, S. Jarecki, H. Krawczgk, and T. Rabin, *"Robust threshold DSS signatures", In EUROCRYPT'96,* pp.354-371, 1996. Springer-Vwelag. LNCS No. 1070. 90-104

[9] 안상만, 오수현, 원동호, "다중서버를 이용한 패스워드 기반 키분배 방식에 관한 연구", 한국정보보호학회 종합학술발표회 논문집 Vol. 12, No. 1

[10] Mario Di Raimondo and Rosario Gennaro, "Probably Secure Threshold Password-Authenticated Key Exchange," *In EUROCRYPT'2003, LNCS 2656,* pp.507-523, 2003.

[11] R. Canetti, R Gennaro, S. Jarecki, H. Krawczyk and T. Rabin, *"Adaptive Security for Threshold Cryptosystems," In CRYPTO'99 Springer LNCS1666,* pp.98-115, 1999.

저 자 소 개

이 영 숙(Young-Sook Lee) 준회원

- 1987년 2월 : 성균관대학교 정보공학과(공학사)
- 2002년 9월~현재 : 성균관대학교 정보통신대학원 정보보호학과(석사과정)
- 2002년 3월~현재 : 두원공과대학 소프트웨어개발과 강사

<관심분야> : 정보통신 보안, 암호 알고리즘,

이 영 교(Young-Gyo Lee) 정회원

- 1986년 2월 : 한양대학교 전자공학과(공학사)
- 1991년 6월 : 한양대학교 전자공학과 일반대학원(공학석사)
- 2002년 3월~현재 : 성균관대학교 정보통신공학부 박사수료
- 1993년 3월~1998년 9월 : 대우통신종합연구소 선임연구원
- 1999년 2월~2001년 6월 : LG전자/정보통신 중앙연구소 선임연구원
- 2002년 3월~현재 : 인하공업대학 정보통신과 초빙교수

<관심분야> : 정보통신 보안, 네트워크 보안

원 동 호(Dong-Ho Won) 정회원

- 성균관대학교 전자공학과(학사, 석사, 박사)
- 한국전자통신연구소(ETRI) 전임연구원
- 일본 동경공업대 객원 연구원
- 성균관대학교 전산소장, 교학처장, 전기전자 및 컴퓨터공학부장, 정보통신대학원장, 정보통신기술연구소장, 연구처장, 국무총리실 국가정보화 추진자문위원회 자문위원, 한국정보보호학회 이사, 부회장, 수석 부회장, 회장(명예회장)
- 현재 : 성균관대학교 정보통신공학부 교수
정통부 지정 정보보호인증기술연구센터 센터장
<관심분야> : 암호학, 정보통신 보안, 암호알고리즘