情報保護學會論文誌
第12券 第5號, 2002. 10

패스워드 기반의 상호 인증 및 키 교환 프로토콜*

박 호 상**, 정 수 환**

A password-based mutual authentication and key-agreement protocol

Ho-Sang Park**, Souhwan Jung**

요 약

본 논문에서는 기존에 제안된 패스워드 기반의 프로토콜 중 공개키 기반의 프로토콜을 분석하고, 사용자의 서명을 이용한 패스워드 기반의 상호인증 및 세션키 교환 프로토콜인 AKE-ECC를 제안하였다. 제안 프로토콜 AKE-ECC는 사용자와 서버에서 타원곡선의 점의 곱셈 회수가 각각 2회가 실행되며, 서명생성과 인증을 위한 키 쌍(공개키, 비밀키) 생성과 세션키 생성에서 기존의 프로토콜과 달리 키 교환 알고리즘(ECDH)을 이용하여 생성하였다. 또한, 패스워드 기반 프로토콜의 취약점인 dictionary attack 등 알려진 여러 공격으로부터 안전하며, 제안 프로토콜의 안전성은 ECDH, ECDLP를 기반으로 한다.

ABSTRACT

This paper proposes a password-based mutual authentication and key agreement protocol, which is designed by applying ECDSA and ECDH. The proposed protocol, AKE-ECC, computes 2 times of point multiplication over ECC on each of client and server, and generates the key pairs(public key, private key) and a common session key using ECDH that is different compare to previously proposed protocol. It is against common attacks include a dictionary attack and the security of proposed protocol is based on the ECDLP, ECDH.

Keyword : *ECDSA, ECDLP, ECDH, Password-based key agreement, Mutual authentication*

Ⅰ. 서 론

안전하지 못한 공중 네트워크에서 정당한 사용자와 그렇지 않은 사용자에 대한 구별이 되어야 하며, 정당한 사용자와의 통신에서는 정보 보호를 위한 안전한 통신 채널 설정이 요구된다. 안전한 통신 채널은 두 통신자간의 상호인증과 세션키를 이용한 암호화 통신으로써 가능하며, 안전한 보안 프로토콜의 개발과 동시에 효율성, 적용의 용이성, 사용자의 편리성 등을 고려해야 한다. 안전한 통신 채널 형성을 위한 인증 방법에는 여러 가지가 있지만, 사람의 지식을 이용하는 방법이 가장 효율적이며, 패스워드가 대표적이다. 패스워드는 송·수신 메시지 암호화를 위한 암호화 키, 그룹의 생성자(generator), 그리고 송·수신 메시지의 연산시 지수 등으로 사용되며, 이들 패스워드 정보는 DH(Diffie-Hellman) 알고리즘 또는 공개키 기반의 암호알고리즘 등에 적용되어 세션키 생성 및 상호인증 등의 기능을 수행하게된다.

키 교환 알고리즘으로 가장 널리 알려진 DH는 DLP(Discrete Logarithm Problem)를 기반으로 하고 있다. 키 생성 알고리즘인 DH에 인증정보를

* 본 연구는 학술진흥재단 협동연구과제(과제번호 2001-042-E00045) 지원으로 수행하였습니다.
** 숭실대학교 정보통신전자공학부(hosang@cns.ssu.ac.kr , souhwanj@cns.ssu.ac.kr)

추가하여 공격자로부터 안전성을 제공한다. 대표적인 프로토콜들로는 A-EKE[8], B-SPEKE[7], SRP[5], PAK-X[1,4], AMP[3]등이 있으며, 이들 프로토콜들의 통신회수, 연산량 등을 간략하게 살펴보면 다음과 같다.

A-EKE[8]는 사용자와 서버간에 각각 2회의 지수연산을 실행하지만, 메시지를 암호화하여 주고받음으로써 암·복호화를 위한 추가 시간이 필요하게 되고, 통신회수는 5회로서 비교적 많은 통신회수가 필요하다. B-SPEKE[7]는 4회의 통신회수가 소요되며, 사용자와 서버간에 각각 3회, 4회의 지수연산이 실행된다. SRP[5]는 패스워드 파일을 비대칭으로 저장하여 파일이 노출되었을 때 패스워드가 직접 노출되지 않게 하고, 네트워크 상에서 어떠한 패스워드의 정보를 유출시키지 않는 영지식(zero-knowledge)을 지향하지만, 4회의 통신회수가 소요되고, 서버와 사용자는 각각 3회의 지수연산이 실행된다. AMP[3]는 사용자와 서버에서 각각 2회의 지수연산을 실행하며, 소요되는 통신회수는 4회이다. PAK-X[1,4]는 다른 프로토콜에 비해 통신회수를 1회 줄여 3회가 소요되지만, 지수연산은 서버와 사용자에서 각각 4회 실행으로 비교적 많은 회수가 실행된다. PAK-X의 많은 연산량을 간결하게 하기 위해 PAK-EC[10]가 제안되었으며 통신회수는 3회를 유지하였지만 연산량은 PAK-X에 비해 감소하였다. 여기에 사용된 연산은 ECDH(Elliptic curve DH)로서 사용자와 서버에서 각각 3회를 실행하게 되며, ECDLP에 의해 여러 공격으로부터 안전성을 보장한다. 이들 DH 기반의 프로토콜들의 통신회수나 연산량에 대해서는 Ⅲ.1절에서 표로써 자세히 나타내었다.

공개키 암호알고리즘을 이용한 경우, 가장 널리 사용된 공개키 암호화 알고리즘은 RSA이다. 공개키 암호화 알고리즘은 많은 연산량으로 적용에 부담이 있으며, 대표적인 프로토콜들로는 A-EKE[8], OKE[11], SNAPI-X[9], M-SNAPI-X[13] 등이 있다.

A-EKE[8]는 DH의 방법으로 키를 생성하고 인증을 하게되지만 공개키 암호알고리즘을 이용하여 마지막 단계에서 사용자가 실제 패스워드를 이용하여 세션키를 생성하였는지에 대한 확인 과정으로 RSA서명을 이용하였다. 사용자는 서버에게 서명을 전송하고 서버는 이를 검증하여 실제 패스워드가 사용되었음을 확인한다. 하지만 여기서 사용된 키 쌍은 매 세션마다 일정한 값이다. SNAPI-X[13]는 RSA 암·복호화 알고리즘을 이용하여 사용자를 인증하게 되는 알고리즘으로서, 지수연산을 서버와 사용자에서 각각 2회, 1회로 줄였으나, 통신회수는 5회를 실행한다. 공개키 기반의 프로토콜에 대한 통신회수와 연산량에 대한 비교는 Ⅲ.1에서 표로써 자세히 나타내었다.

본 논문에서는 기존에 제안된 프로토콜 중 공개키 기반의 프로토콜을 분석하고 서명기법을 도입하여 사용자의 세션키에 대한 부인방지를 제공하며, 또한 인증에 사용되는 키 쌍과 세션키는 두 통신자간의 상호 키 교환 알고리즘에 따라 생성하도록 하였다. 논문은 제안하는 프로토콜에 대하여 설명한 Ⅱ장, 제안 프로토콜의 특징과 안전성에 대하여 분석한 Ⅲ장 그리고, Ⅳ장의 결론으로 구성되었다.

Ⅱ. 제안 프로토콜

이번 장에서는 본 논문에서 제안하는 프로토콜 AKE-ECC에 대해 설명하였다. AKE-ECC는 공개키 알고리즘인 ECDSA(EC Digital Signature Algorithm)를 적용하여, 키 교환으로 생성된 세션키에 대해서 서명을 통하여 인증하게 하고, 통신회수는 4회로서 전체적인 구조는 [Fig. 1]과 같다. 기존의 공개키 기반의 프로토콜들 OKE,SNAPI-X, A-EKE 등은 암호알고리즘의 연산시간이 많이 소요되는 RSA를 적용하였으며, 또한 세션키 생성에 있어서는 DH를 이용하지 않고 사용자 또는 서버에 의하여 임의로 생성한 값을 상호 인증에 사용하여 세션키로 공유하였다. 또한 인증에 사용되는 키 쌍도 사용자 또는 서버에 의해 임의로 선택되고 공개키를 상대방에게 전송하여 이를 이용하게 하였다. A-EKE 경우에는 사용되는 키 쌍이 매 세션마다 같은 값이 사용되는 단점이 있다. 기존의 공개키 기반의 프로토콜과 비교하여 AKE-ECC의 특징은, 공개키 알고리즘에 사용되는 키 쌍은 키 교환 알고리즘(ECDH)을 통해 생성하여 인증에 사용하며, 세션키도 키 교환 알고리즘(ECDH)에 의하여 생성한다. 세션키 인증은 세션키에 대한 부인방지 등을 제공하는 사용자의 서명을 이용하였으며, 전체 공개키 암호알고리즘의 연산시간의 효율성을 고려하여 사용자가 서명을 생성하고, 서버가 검증하였으며 타원곡선 알고리즘을 적용하였다. 또한 dictionary attack등 알려진 여러 공격으로부터 안전성을 제공한다. AKE-ECC의 특징과 안전성에 대해서는 Ⅲ장에서 자세하게 설명하였다.

A(client)		B(server)
$a \in_R F_q$		
$(a \cdot H_1(\pi)) \cdot G = (x_a, y_a)$		
	(x_a, y_a) →	$b \in_R F_q$
		$(b \cdot H_1(\pi)) \cdot G = (x_b, y_b)$
	(x_b, y_b) ←	
$k = (a \cdot b \cdot H_1(\pi) \cdot G)(x)$		$k' = (b \cdot a \cdot H_1(\pi) \cdot G)(x)$
$d = k \cdot a \cdot H_1(\pi)$		
$M = H_2(k)$		$dG = k' \cdot (x_a, y_a)$
$\delta \in_R F_q$		
$\delta \cdot G = R$		
$r = x_R$		
$s = \delta^{-1}(H_3(M) + dr)$		
	r, s, M →	
		$w = s^{-1}$
		$u_1 = H_3(M) \cdot w,\ u_2 = r \cdot w$
		$u_1 G + u_2 \cdot (dG) = (x_R, y_R)$
		$v = x_R$
		$verify:\ v \stackrel{?}{=} r$
$verify: H_4(k') \stackrel{?}{=} H_4(k)$	$H_4(k')$ ←	$SK = H_5(k')$
$SK = H_5(k)$		

[Fig. 1] AKE-ECC

전체적인 AKE-ECC를 설명하면, 서버는 사용자의 패스워드를 사전에 공유하며, 사용자가 서버와 통신을 원하면, 우선 임의의 값을 선택하고 자신의 패스워드정보 값을 이용하여 타원곡선위의 임의의 점을 계산하여 결과를 서버에게 전송한다. 서버는 수신한 메시지에서 공유한 패스워드정보를 이용해 임의의 점을 추출하며, 임의의 생성 값과 공유한 패스워드 정보를 이용하여 타원곡선상의 임의의 점을 생성하여 결과 값을 전송한다. 전송후 서버는 ECDH에 의해 세션키와 사용자의 공개키를 생성한다. 사용자는 ECDH를 이용하여 세션키를 생성하고 자신의 비밀키를 이용하여 세션키에 대한 서명으로서 인증정보를 생성하고 결과를 서버에게 전송한다. 서버는 이전에 생성한 사용자의 공개키를 이용하여 메시지인 세션키를 검증하게 되고 사용자를 인증하게 된다 (검증이 되지 않는 경우 실행을 중단한다.). 인증후 서버는 자신의 인증을 위해 세션키를 해쉬하여 사용자에게 전송함으로써 사용자는 결과를 비교하여 일치할 경우 서버를 인증하고 공통의 세션키를 생성한다. AKE-ECC는 [Fig. 1]과 같다.

- Notation :

A, B : 통신자의 ID
π : 공유한 패스워드
SK : A와 B에 의하여 생성된 세션키
α, β : 타원곡선 $y^2 = x^3 + \alpha x + \beta$를 정의하는 계수 $\alpha, \beta \in F_q$ (도메인파라미터)
$\# E$: 타원곡선 위의 점의 개수
q : 유한체의 크기 (도메인파라미터)
n : basepoint G의 위수(order) (도메인파라미터)
h : $\# E/n$ 공통인자 (도메인파라미터)
G : 타원곡선의 basepoint(위수가 n인 그룹의 점의 생성원) (도메인파라미터)
H : 일 방향 해쉬함수

- 설정단계 :

A와 B는 패스워드 π와 타원곡선 도메인 파라미터 $(\alpha, \beta, q, n, h, G)$를 공유한다. B는 A의 패스워드 자체를 패스워드 파일에 저장한다.

- 실행단계 :

① A는 임의의 값 a를 선택하고, 자신의 패스워드를 이용하여 $H_1(\pi)$를 계산하고, 타원곡선의 basepoint

G를 이용하여 $(a \cdot H_1(\pi) \cdot G)=(x_a, y_a)$를 계산한다. A는 이 결과 값을 B에게 전송한다.

② ①의 메시지를 받은 B는 임의의 값 b를 선택하고 자신의 패스워드를 이용하여 $H_1(\pi)$를 계산하고, 타원곡선의 basepoint G를 이용하여 $(b \cdot H_1(\pi) \cdot G)=(x_b, y_b)$를 계산한다. 이 결과 값을 A에게 전송후 B는 ECDH에 의해 $k'=[b \cdot a \cdot H_1(\pi) \cdot G](x)$를 계산하고, A의 공개키인 $dG=k' \cdot (x_a, y_a)$를 계산한다.

③ ②의 메시지를 받은 A는 $k=[a \cdot b \cdot H_1(\pi) \cdot G](x)$를 계산하고, 비밀키 $d=k \cdot a \cdot H_1(\pi)$를 계산한다. 비밀키 d를 이용하여 메시지 M에 대한 서명 (r, s)를 생성한 후 B에게 전송한다. 여기서 $M=H_2(k)$이다.

④ ③의 메시지를 받은 B는 A의 공개키를 $dG=k' \cdot (x_a, y_a)$를 이용하여 A의 서명을 검증한다. 서명이 검증되면, B는 A를 인증하고, $H_4(k')$를 계산하고, A 에게 전송한다. B는 전송후 세션키 $SK=H_5(k')$를 생성한다.

⑤ ④의 메시지를 받은 A는 받은 메시지와 자신이 계산한 메시지, $H_4(k)=H_4(k')$ 를 비교하여 일치할 경우 A는 B를 인증 한다. 인증후 A는 세션키 $SK=H_5(k)$를 생성한다.

AKE-ECC의 프로토콜은 [Fig. 1]과 같으며 특징과 안전성에 대해서는 Ⅲ장에서 설명하였다.

Ⅲ. 안전성 분석

이번 장에서는 AKE-ECC와 기존 프로토콜을 비교하여 특징과 안전성에 대한 분석을 하였다. I장 서론에서 설명한 기존 프로토콜과 비교하여 설명하였으며, 여러 가지 공격 방법들로부터 AKE-ECC의 안전성을 설명하였다.

3.1 AKE-ECC의 특징

이번 절에서는 지금까지 소개되었던 주요 프로토콜과 [Fig. 1]의 AKE-ECC를 비교하여 장점을 알아보았다. 비교 내용으로는 프로토콜의 통신회수와 연산량, 키 생성방법 등이다. 프로토콜은 크게 공개키 기반의 프로토콜과 DH 기반의 프로토콜로 나눌 수 있음을 I장에서 설명하였다.

DH를 이용한 프로토콜로는 A-EKE[8], B-SPEKE[7], SRP[5], PAK[1,4], PAK-EC[10], AMP[3] 등이다. 사용자와 서버간의 통신회수를 알아보면, 소요되는 회수는 각각 3, 4, 5회이며, PAK 는 다른 프로토콜과 비교할 때 같은 수준의 안전도를 제공하면서 통신 회수를 3회로 줄여 통신회수에서 장점을 가진다. 프로토콜에서 연산 시간과 관련이 있는 것은 지수연산이다. 지수연산을 보면 PAK-EC는 사용자와 서버에서 각각 3회씩 실행하며,

AMP에서는 각각 2회씩 실행함으로 지수 연산에서는 AMP가 PAK-EC보다 장점을 갖는다. PAK-EC에서 지수연산은 타원곡선에서 점의 곱셈연산을 의미한다.

공개키 기반의 프로토콜에서 A-EKE[8]는 RSA 서명을, 그리고 OKE[11], SNAPI-X[9]는 RSA의 암·복호화를 적용하였으며, AKE-ECC는 ECDSA 기법을 적용하였다. 프로토콜의 통신회수를 볼 때, AKE-ECC와 OKE가 4회로서 최소이다. 지수연산은 각 암호화 알고리즘, 또는 서명에서 실행되는 연산을 제외한 것으로 OKE는 실행되지 않으며, 제안 프로토콜은 사용자와 서버에서 각각 2회를 실행한다. 하지만, OKE는 공격의 취약점이 있음이 증명되었으므로[13], SNAPI-X가 최소로 실행된다. AKE-ECC에서 실행하는 지수연산은 타원곡선에서 점의 곱셈연산을 의미한다. 네트워크 상에서의 통신 회수는 네트워크 자원의 효율성과 네트워크상의 지연(delay) 등을 고려할 때 통신회수가 적을수록 장점이 있다. 그러나 통신 회수가 줄어도 보안의 안전도는 변함이 없어야 한다. SNAPI-X, OKE는 사용자 또는 서버에 의해 임의로 인증키와 세션키가 생성, 선택되어지지만 제안 프로토콜은 DH 키 교환 알고리즘에 의해 생성되며, A-EKE의 키 쌍은 패스워드로부터 생성되어 패스워드 파일에 저장된 값이므로 항상 일정한 값을 갖는다.

AKE-ECC는 사용자의 서명을 이용함으로써 사용자의 부인방지를 제공하고, 인증에 사용되는 키 쌍은 두 통신자 사이의 키 교환에 의해 생성된다. [Fig. 1]을 이용하여 설명하면, $r=x_R$, $s=\delta^{-1}(H_3(M)+dr)$는 사용자에 의해 생성되는 서명 값으로 사용자가 자신의 비밀키 $d=k \cdot a \cdot H_1(\pi)$ 로서 서명 값 r, s를 생성하고, 서버는 해당하는 공개키 $dG=k' \cdot (x_a, y_a)$로서 서명을 검증하여 인증하게 됨으로 생성한 세션키 k에 대한 사용자의 부인을 방지할 수 있다. OKE, SNAPI-X 프로토콜은 사용자의 인증과 세션키 생

〔Table 1〕 on-line 에서 pass 및 계산량 비교

구분	프로토콜	pass	암호화		인증키 생성	세션키 생성	지수연산		random 생성	
			client	server			client	server	client	server
공개키	A-EKE	5	RSA서명		키쌍 일정	DH	2	2	2	2
	SNAPI-X	5	RSA		키쌍 임의선택	임의선택	1	2	2	2
	OKE	4	RSA		키쌍 임의선택	임의선택	x	x	1	2
	AKE-ECC	4	ECDSA		DH 키 교환	DH	2	2	1	1
DH기반	A-EKE	5	3	3	x	DH	4	4	1	1
	B-SPEKE	4	x	x	x	DH	3	4	1	2
	SRP	4	x	x	x	DH	3	3	1	1
	AMP	4	x	x	x	DH	2	2	1	1
	PAK-X	3	x	x	x	DH	4	4	1	2
	PAK-R	3	x	x	x	DH	3	3	2	1
	PAK-RY	3	x	x	x	DH	4	5	3	1
	PAK-EC	3	x	x	x	DH	3	3	1	1

성에 있어 사용자 또는 서버가 임의의 키 쌍을 생성하여 인증을 하였으며, 세션키 생성에도 임의로 선택된 값을 세션키값으로 공유한다. 제안 프로토콜에서는 사용자의 인증에 사용된 키쌍은 사용자와 서버의 키 교환 알고리즘(ECDH)에 의해 매 세션마다 새롭게 생성되는 값이며, 세션키도 ECDH에 의해 매 세션마다 생성된 값이다. 설명한 것과 같이 제안 프로토콜은 인증방법과 인증 키 쌍의 생성 및 세션키 생성이 기존의 프로토콜과 다른 방법을 이용하였다. dictionary attack등 여러 공격으로부터의 안전성은 안전성 분석에서 설명하였다.

3.2 안전성 분석

AKE-ECC는 ECC, ECDLP, ECDH의 안전성을 기반으로 한다. ECC 자체의 안전성에 대해서는 여러 논문과 표준[12]등에서 증명하였으므로 본 논문에서는 ECC의 안전성에 대한 설명은 제외하며, 패스워드 기반의 프로토콜이 취약한 공격만을 자세하게 설명한다.

- Dictionary attack

Dictionary attack은 공격자가 사용자의 패스워드를 추측하여 추측한 패스워드를 실제 메시지에서 드러나는 값에 대입하여 결과를 비교하여 실제 패스워드를 찾는 공격 방법이다. AKE-ECC에서는 사용자가 자신의 메시지를 전송하고 서버로부터 받는 메시지는 첫 번째 메시지와 상관없는 값을 받게 됨으로 추측한 패스워드로서 생성한 결과 값을 비교할 수 없다. 따라서 dictionary attack은 불가능하다. 또한 세 번째 메시지를 받은 서버는 정당한 패스워드 정보를 사용하는 사용자만이 인증하게 됨으로 더 이상 dictionary attack은 불가능하다.

- Replay attack

공격자가 사용자의 메시지를 재 전송하여 이미 정상적인 사용자에 의해 생성된 이전 키(old session key)를 다시 생성하기 위함이다. 이 공격 방법은 사용자와 서버간에 항상 임의의 값을 사용하기 때문에 불가능하다. AKE-ECC에서는 사용되는 값인 a와 서버에 의해 생성되는 임의의 값인 b가 매 세션마다 새롭게 생성됨으로써 반복(replay)에 의한 이전 키 생성이 불가능하다. 따라서 AKE-ECC는 replay attack으로부터 안전하다.

- PFS(Perfect Forward Secrecy)

PFS는 "현재의 세션키 정보가 알려져도 이전키를 알 수 없다"는 것으로 PFS를 제공하기 위해서는 사용자와 서버간에 주고받는 메시지 내용이 이전 키 생성을 위한 정보와 관련이 없어야 한다. 이를 위해서는 생성되는 세션키값이 항상 임의의 값으로 생성되어야 하고, 메시지에서 임의의 값이 ECDLP에 의해 보호되어야 한다. AKE-ECC의 세션키값은 a, b로서 생성되고, ECDLP에 의해 알려지지 않는다. a, b값이 알려지지 않음으로 AKE-ECC는 PFS를 제공한다.

• Denning-Sacco attack[3]

Denning-Sacco attack은 이전키를 안다고 할 때 패스워드를 알아내는 공격 방법이다. AKE-ECC의 세션키값은 세션마다 임의의 값 a,b로서 생성되며 a,b값은 ECDLP에 의해 분리할 수 없으므로, 패스워드가 알려져도 이전의 세션키를 알 수가 없다.

• MITM(Man-In-The-Middle) attack

MITM attack은 공격자가 사용자와 서버사이에 존재하여 사용자와 서버의 메시지를 가로채어 사용자와 공격자, 공격자와 서버간에 각각의 세션키를 생성하는 공격방법이다. AKE-ECC에서는 인증 정보인 패스워드가 사용되고 있으므로 공격이 불가능하다. 따라서 MITM attack이 불가능하다.

• Impersonation attack

서버의 패스워드 파일이 노출되었을 때의 공격방법으로 AKE-ECC에서는 패스워드 자체가 직접 저장됨으로 impersonation attack을 막지 못한다. 그리고, 패스워드가 직접 저장되지 않고 검증자가 저장되어도 검증자 자체가 dictionary attack으로 공격이 가능하기 때문에 impersonation attack은 고려할 의미가 없다.

지금까지 제안 프로토콜인 AKE-ECC에 대한 여러 가지 공격방법에 대해서 알아보았다. 설명한 것과 같이 알려진 여러 공격으로부터 안전하다.

Ⅳ. 결 론

본 논문에서는 향후 널리 사용될 것으로 전망되는 패스워드를 기반으로 하며 공개키 알고리즘을 이용한 사용자와 서버간의 상호인증과 세션키 교환 프로토콜인 AKE-ECC를 제안하였다. AKE-ECC는 서명을 이용하여 사용자를 인증하게 됨으로써 사용자의 세션키에 대한 부인방지 기능을 제공하며, 서명에 사용되는 키 쌍의 생성과 세션키의 생성은 모두 ECDH에 의한 키 교환 알고리즘에 의해 생성된다. 사용자와 서버에서 타원곡선의 점의 곱셈 회수는 각각 2회가 실행된다. 따라서 AKE-ECC는 지금까지의 공개키 기반의 프로토콜, A-EKE, OKE, SNAPI-X 등이 갖는 특징인 일정한 인증키 값 사용, 임의의 통신자에 의한 세션키 선택과 암·복호화 키 생성 방법 등에서 차이점이 가지며, 패스워드 기반 프로토콜의 알려진 여러 가지 공격에 대해 안전성을 갖는다. ECDSA는 서명 생성 시간이 검증 시간 보다 짧아 무선 단말기를 이용한 WPKI 적용에 적합한 특성을 가지므로 제안 프로토콜 AKE-ECC는 사용자의 서명 생성에 소요되는 연산량을 줄이는 장점을 갖는다.

참 고 문 헌

[1] V. Boyko, P. MacKenzie & S. Patel, "Provably Secure Password Authenticated Key Exchange Using Diffie-Hellman," Advances in Cryptology - EUROCRYPT 2000, Preneel, B., (Ed.), May 14-18, 2000.

[2] D. Jablon, "Strong Password-Only Authenticated Key Exchange," Computer Communication Review, ACM SIGCOMM, Vol. 26, No. 5, pp. 5~26, October 1996.

[3] T. Kwon, "Authentication and Key Agreement via Memorable Passwords," NDSS 2001 Symposium Conference Proceedings, February 7-9, 2001.

[4] P. MacKenzie, "On the Security of the SPEKE Password-Authenticated Key-Exchange Protocol," Cryptology ePrint Archive: Report 2001/057

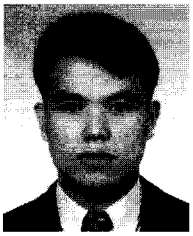
[5] T. Wu, "The Secure Remote Password Protocol," Proceedings of the 1998 Internet Society Network and Distributed System Security Symposium, San Diego, pp. 97~111, March 1998.

[6] S. M. Bellovin and M. Merritt, "Encrypted Key Exchange: Password- Based Protocols Secure Against Dictionary Attacks," Proceedings of the I.E.E.E. Symposium on Research in Security and Privacy, Oakland, May 1992.

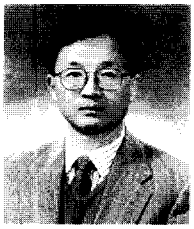
[7] D. Jablon, "Extended Password Key Exchange Protocols Immune to Dictionary Attacks," Proceedings of the Sixth Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET-ICE '97), IEEE Computer Society, Cambridge, MA, pp. 248~255, June 18-20,

1997.
[8] Steven M. Bellovin, Michael Merritt, Augmented Encrypted Key Exchange: A Password-Based Protocol Secure against Dictionary Attacks and Password File Compromise," ACM Conference on Computer and Communications Security 1993: pp. 244~250.
[9] P. MacKenzie & R. Swaminathan, "Secure Network Authentication with Password Identification," Presented to IEEE P1363a, August, 1999.
[10] P. MacKenzie, "More Efficient Password-Authenticated Key Exchange," Springer-Verlag, LNCS 2020, pp. 361~377, April 8-12, 2001.

〈著者紹介〉



박 호 상 (Ho-Sang Park) 정회원
1997년 2월 : 시립인천대학교 물리학과 학사
2000년~현재 : 숭실대학교 대학원 정보통신공학과 석사과정
〈관심분야〉 사용자 인증, Key Management, PKI



정 수 환 (Souhwan Jung) 정회원
1985년 2월 : 서울대학교 전자공학과 학사
1987년 2월 : 서울대학교 전자공학과 석사
1988년~1991년 : 한국통신 전임연구원
1996년 : 미 워싱턴 주립대(시애틀) 박사
1996년~1997년 : Stellar One SW Engineer
1997년~현재 : 숭실대학교 정보통신전자공학부 조교수
〈관심분야〉 VoIP security, 사용자 인증, 네트워크 프로토콜 보안