情報保護學會論文誌
第18卷 第2號, 2008. 4

서버의 개입이 없는 스마트카드 기반의 3자간 키 교환 프로토콜*

김 용 훈[1†], 윤 택 영[1], 박 영 호[2‡]
[1]고려대학교, [2]세종사이버대학교

Smart card based three party key exchange protocol without server's aid*

Yong Hun Kim[1†], Taek-Young Youn[1], Young-Ho Park[2‡]
[1]Korea University, [2]Sejong Cyber University

요 약

3자간 키 교환 프로토콜은 신뢰할 수 있는 서버의 도움을 받아 서버와 공유한 패스워드를 사용하여 사전에 특정 정보를 공유하지 않은 사용자와의 인증된 키 교환을 수행할 수 있는 암호학적 기법이다. 사용자는 다수의 패스워드를 기억하지 않고 서버와 공유한 패스워드만 기억하면 다수의 사용자와 키 교환이 가능하다. 서버는 사용자들의 키 교환 프로토콜이 수행될 때마다 두 사용자를 인증해주기 위한 통신과 연산을 수행한다. 따라서 동시에 다수의 사용자가 키 교환 프로토콜을 수행하면 서버에 큰 부담이 된다. 본 논문에서는 3자간 키 교환 프로토콜에서 실제 키 교환 수행 시 서버가 개입하지 않음으로서 서버의 부담을 최소화하는 프로토콜을 제안한다. 제안하는 프로토콜에서는 사용자의 등록 시에만 서버가 개입하고, 실제 키 교환 수행과정은 두 사용자들 사이에서만 이루어지므로 키 교환과정에서 서버는 연산이나 통신을 수행하지 않는다.

ABSTRACT

Three-party key exchange protocol is a cryptographic protocol which permits two clients share a common session key using different passwords by the help of a trusted server. In a three-party key exchange protocol, an user remember only one password which shared with a trusted server for establish a common key with another user. The trusted server should participate in an execution of the protocol between two clients. This impose heavy burden on the server when many users want to establish a session key using the protocol. In this paper, we propose a three-party key exchange protocol based on a smart card which reduce the computational complexity and communication overhead for the trusted server. In our protocol, the server does not participate in an key exchange procedure between two clients.

Keywords : key exchange protocol, smart card, password

Ⅰ. 서 론

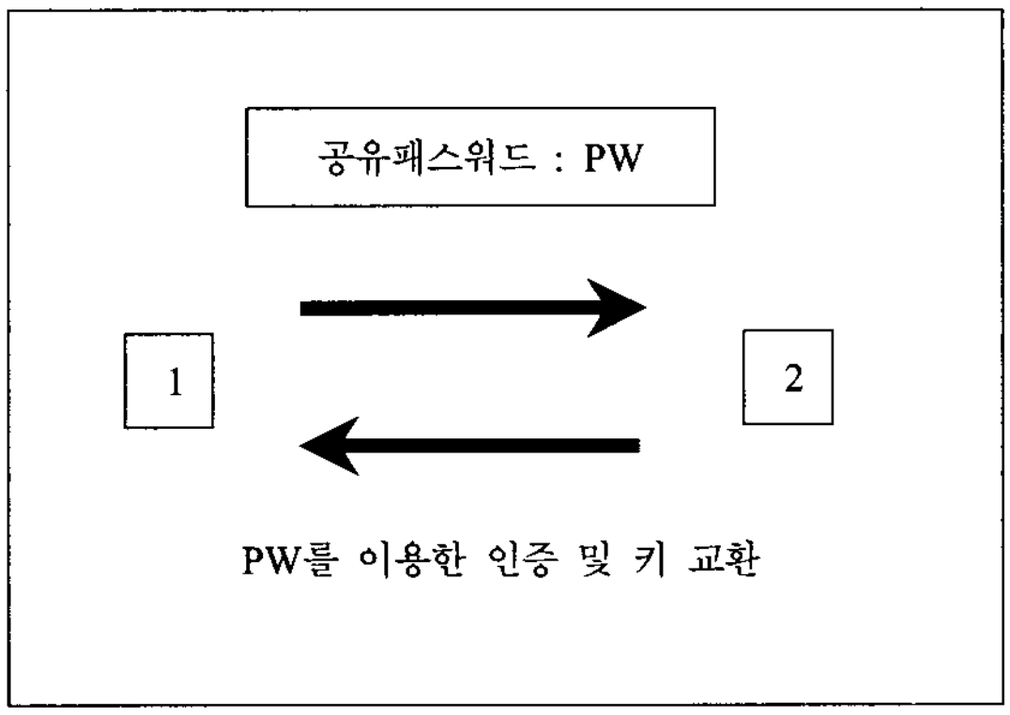
키 교환 프로토콜은 안전한 통신을 위한 중요한 기술 중 하나로서 안전하지 않은 채널에서 키를 공유하기 위한 암호학적 도구이다. 키 교환 프로토콜은 1976년

접수일: 2007년 11월 12일; 채택일: 2008년 1월 15일
* 이 논문은 2005년 정부(과학기술부)의 재원으로 학국과학재단의 지원을 받아 수행된 연구임(R01-2005-000-11261-0).
† 주저자, yhkim@cist.korea.ac.kr
‡ 교신저자, youngho@sjcu.ac.kr

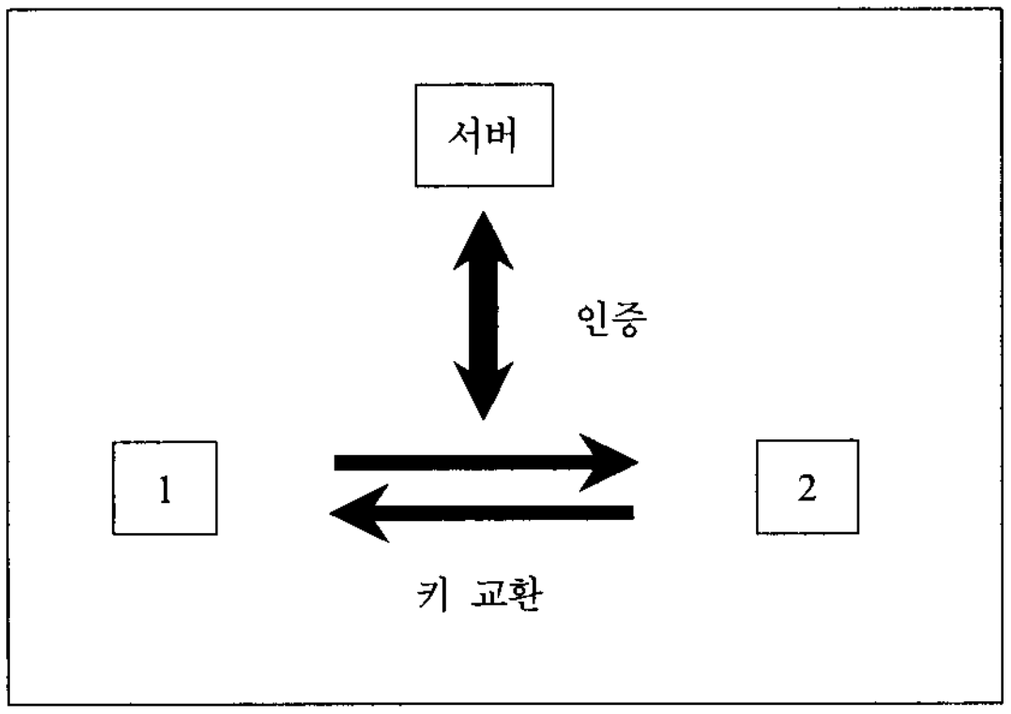
Diffie와 Hellman[9]에 의해 처음으로 제시되었다. 그러나 이 방법은 상대방에 대한 인증이 고려되지 않았기 때문에 많은 약점이 발견되었다. 그 후 사용자 인증을 제공하는 키 교환 프로토콜에 대한 많은 연구가 진행되었다. 키 교환 프로토콜에서 사용자 인증을 위한 방법은 다음과 같이 크게 3가지로 분류할 수 있다. 공개키 암호 시스템을 이용하는 방법, 사전에 공유된 키를 사용하여 대칭키 암호 시스템을 이용하는 방법, 패스워드를 사용하는 방법이 있다. 공개키 암호 시스템을 이용하는 방법은 신뢰 할 수 있는 기관인 인증서 발급 기관이 발급한 인증서로 상호간의 인증을 수행한 후 인증 받은 공개키를 기반으로 키를 설정하는 방법이고, 대칭키 암호 시스템을 이용하는 방법은 프로토콜 수행 전에 상호간에 공유된 키를 사용해 사용자 인증과 키 설정을 하는 방법이다. 암호 시스템에서 사용할 수 있는 키는 보통 길이가 긴 난수로서 사람이 기억하기 힘들다. 따라서 두 방법 모두 암호 시스템에서 사용할 수 있는 키를 사용자가 저장하고 있어야 할 저장매체가 필요하다. 키 교환 프로토콜에서 사용자 인증을 위한 또 다른 방법은 패스워드를 이용한 방법이다. 인증을 위해 사용되는 패스워드는 사람이 기억하기 쉬운 비교적 짧은 문자열이므로 패스워드를 저장할 특별한 저장매체나 컴퓨터의 메모리가 필요 없다. 이는 공개키나 사전에 공유된 키를 사용하는 방법과 비교해서 사용자의 측면에서 편리하게 사용할 수 있는 방법으로 많은 연구가 진행되고 있다.

가장 기본적인 패스워드 기반의 키 교환 프로토콜은 Bellovin과 Merritt(1992)[3]가 처음 제안하고 그 이후로도 많은 연구가 되어온 2자간 키 교환 프로토콜(그림 1)이 있다[4,5,12,16]. 2자간 키 교환 프로토콜은 서로 공유하고 있는 패스워드를 사용해 상대방을 인증하고 키를 교환하는 방법이다. 2자간 키 교환 프로토콜은 키를 공유하고자 하는 사용자끼리 패스워드를 공유하고 있어야 하므로 키 교환 프로토콜을 수행하기 전에 상대방과 패스워드를 교환해야 하고 또한 다수의 사용자와 키를 교환할 경우 사용자 개인이 다수의 패스워드를 기억하고 있어야 한다. 일반적으로 사람이 기억할 수 있는 패스워드의 개수는 한정되기 때문에 이 방법은 비효율적일 수 있다.

Gong et al. (1993)[10], Halevi와 Krawczyk (1999)[11] 등에 의해 제안된 3자간 키 교환 프로토콜(그림 2)에서는 2자간 키 교환 프로토콜에서 다수의 사용자와 키 교환을 수행하기 위해 많은 패스워드를 기억해야 하는 단점이 개선되었다. 3자간 키 교환 프로토콜은 각 사용자들이 서버와 패스워드를 공유하고 키 교환 수행시 공유한 패스워드를 사용해 각 사용자들이 서버에 의해 인증을 받은 후 인증정보로 상호간의 키를 공유하는 방법이다. 따라서 사용자들은 서버와 공유한 자신의 패스워드 하나만 기억하고 있으면 서버에게 자신의 인증을 받을 수 있으므로 자신의 패스워드 하나로 다수의 사용자들과의 키 교환이 가능하다. 기존의 3자간 키 교환 프로토콜에서 서버는 각 사용자들의 키 교환 프로토콜 수행 때 사용자들의 인증을 위해 매번 개입해 연산을 수행한다. 따라서 다수의 사용자가 동시에 키 교환 프로토콜을 수행하면 서버는 각 사용자들을 위한 연산을 처리해 주어야 하고 이는 서버에게 부담이 된다. 또한 이 경우 서버는 많은 연산량 및 통신량으로 인해 과부하가 걸려 사용자들이 서비스를 받기 위해 기다리는 시간이 길어질 수도 있다. 따라서 3자간 키 교환 프로토콜의 효율성을 증가시키는 것은 중요한 문제이고 문제를 해결하기 위한 연구가 계속 되고 있다[6,18,19].



〔그림 1〕 2자간 키 교환 프로토콜



〔그림 2〕 3자간 키 교환 프로토콜

최근에 스마트카드를 활용하는 분야가 점차 증가하고 있다. 주로 사용되고 있는 교통카드나 신용카드 외에도 회사와 같은 단체에서 내부자의 관리나 인증을 위한 도구로서도 많이 사용되고 있다. 이러한 환경에서도 키 교환이 필요하므로 스마트카드를 사용해 3자간 키 교환 프로토콜의 효율성을 증가시키려는 방법 또한 활발히 연구되고 있다[1,13]. 스마트카드를 사용하는 프로토콜은 서버의 패스워드 저장 비용을 줄일 수 있고 기존의 프로토콜과 비교해 매우 적은 연산량으로 수행이 가능하다. 하지만 이와 같은 프로토콜 역시 서버의 개입이 필요하므로 최소 3번 이상의 통신이 필요하게 되어 통신량을 줄이지는 못하였고 서버가 키 교환시 연산을 수행하게 된다. 최근에는 PC와 같은 매체들의 연산능력이 증가함에 따라 3자간 키 교환 프로토콜에서 사용자들의 연산능력 역시 증가하고 있다. 따라서 연산량에 비해 통신량을 줄이는 것이 프로토콜의 효율성 증가에 더 큰 영향을 미치게 된다.

본 논문에서는 키 교환 과정에서 서버의 개입이 없어 서버의 연산을 제거하고 통신량을 최소화한 스마트카드 기반의 3자간 키 교환 프로토콜을 제안한다. 기존의 3자간 키 교환 프로토콜에서 서버는 사용자 등록과정과 각 사용자들의 키 교환과정에 참여하여 연산과 통신을 수행하였다. 두 과정 중에서 사용자 등록과정보다는 키 교환과정이 더욱 빈번하게 발생할 것이고 따라서 키 교환시 서버의 개입을 줄일 수 있다면 서버의 부담은 많이 줄어들게 될 것이다. 따라서 본 논문에서 제안하는 프로토콜은 기존의 프로토콜에 비해 서버의 부담이 크게 줄어들게 된다. 서버의 개입이 없이 키 교환을 수행하는 점은 최소의 통신량으로 키 교환을 가능하게 하고 다수의 사용자들이 동시에 키 교환을 수행하더라도 사용자들의 서비스를 받기 위한 대기 시간이 없어짐을 의미한다. 또한 제안하는 프로토콜은 서버가 각 사용자들의 인증을 위한 정보(ex. 패스워드)를 저장할 필요 없도록 구성하였다. 이는 기존의 3자간 키 교환 프로토콜과 비교해 서버의 저장 비용도 줄이는 효과를 가져왔다.

Ⅱ. 트렙도어 해쉬함수 (Trapdoor Hash function)

본 장에서는 제안하는 프로토콜의 패스워드 변경 과정에서 중요하게 사용되는 트렙도어 해쉬함수(trapdoor hash function)[15,22]에 대해 기술한다. 트렙도어 해쉬함수는 키를 사용하는 해쉬함수로서 함수와 해쉬키(HK), 트렙도어키(TK)로 구성되어 있다. 해쉬키는 공개하는 키로서 해쉬값을 계산하는데 사용되고 트렙도어키는 비밀키로서 충돌쌍을 찾기 위해 사용된다. 그러므로 일반적인 해쉬함수와 같이 해쉬값은 누구든지 계산할 수 있으나 정해진 해쉬값에 대한 충돌쌍은 트렙도어키를 소유하고 있는 사람만이 찾을 수 있다. 트렙도어 해쉬함수는 여러 종류가 있지만 본 논문에서는 트렙도어 해쉬함수의 시스템변수가 제안하는 프로토콜의 시스템변수와 동일한 것을 사용하였다. 본 논문에서 사용하는 트렙도어 해쉬함수는 다음과 같다.

Trapdoor hash function based on the Discrete Log Assumption[15]

- 키 생성 알고리즘 : 안전한 소수($p=2q+1$가 되는 소수 q가 존재하는 소수) $p\in\{0,1\}^k$와 위수가 q인 생성자 $g\in Z_p^*$를 선택한다. 또한 랜덤한 값 $\alpha\in_R Z_q^*$를 선택하고 $y=g^{\alpha}\,(\mathrm{mod}p)$를 계산한다. 공개되는 해쉬키는 (p,g,y)이고 비밀키인 트렙도어키는 α이다.
- 해쉬함수 : 해쉬키 $HK=(p,g,y)$에 대하여 해쉬함수 $h_{HK}: Z_q\times Z_q\rightarrow Z_p^*$를 다음과 같이 정의한다:

$$h_{HK}(m,r)=g^m y^r\,(\mathrm{mod}p).$$

여기서 m은 해쉬를 취할 메세지이고 r은 랜덤하게 선택되는 값이다.

위의 해쉬함수는 트렙도어키 $TK=\alpha$를 사용하면 선택한 메세지 m'에 대하여 $h_{HK}(m,r)=h_{HK}(m',r')$이 되는 r'을 찾을 수 있다. 그 값은 다음과 같음을 쉽게 확인할 수 있다:

$$r'=\alpha^{-1}(m-m')+r\ (\mathrm{mod}q).$$

Ⅲ. 공격 방법 및 안전성 개념

본 장에서는 Dictionary attack을 비롯한 기존에 키 교환 프로토콜의 안전성을 분석하기 위해 고려되었던 공격 방법 및 안전성 개념[17,20]에 대해 서술한다.

3.1. Dictionary attack

Dictionary attack은 패스워드를 기반으로 하는 프로토콜에서 가장 많은 고려가 되고 있는 공격 방법으로서 가능한 문자열을 계속 입력해 보고 맞는 패스워드를 알

아내는 공격 방법이다. 기본적인 방법은 추측한 값을 입력해보고 확인하는 방법이므로 전수조사와 같다. 하지만 패스워드는 사용자가 기억하기 쉽도록 선택하므로 의미있는 내용을 포함하는 값들일 가능성이 매우 높다. 이러한 패스워드의 특성으로 인해 전수조사 대상의 앤트로피는 낮아지게 된다. 또한 패스워드는 사용자와 관련된 정보를 사용할 경우가 많으므로 공격자가 사용자와 관련된 정보를 수집할 경우 패스워드의 앤트로피는 더욱 낮아져서 Dictionary attack은 전수조사와 비교해 매우 강력한 공격 방법이 될 수 있다. Dictionary attack은 On-line Dictionary attack과 Off-line Dictionary attack으로 나눌 수 있다. On-line Dictionary attack은 실제 통신상에서 추측한 패스워드를 입력한 후 상대방이나 서버의 반응을 보고 입력값이 맞는지 확인하는 방법이고, Off-line Dictionary attack은 프로토콜의 수행 중에 전송되는 값들에서 패스워드의 확인이 가능한 값들을 도청해 공격자가 오프라인에서 스스로 계산해보며 추측과 확인을 하는 방법이다.

3.2. Insider attack

Insider attack은 3자간 키 교환 프로토콜에서 많이 고려가 되는 공격으로 올바른 경로로 서버에 등록이 된 사용자가 공격하는 것을 말한다. 따라서 공격자는 다른 올바른 사용자와 정상적인 프로토콜을 수행 할 수 있다. 내부 공격자는 올바른 사용자와의 정상적인 프로토콜을 통해 상대방의 비밀값이나 패스워드 또는 서버의 비밀값을 알아내거나 다른 올바른 사용자간의 통신에 개입해 사용자들의 교환된 키를 알아내려는 시도를 한다.

3.3. Impersonation attack

A와 B가 서버로부터 인증을 받은 올바른 사용자라고 가정하자. Impersonation attack이란 공격자가 자신이 B인 것처럼 속이고 A와 프로토콜을 진행하는 것을 말한다. 공격자는 프로토콜을 진행해 A와 키를 교환한 후 A가 B에게 보내는 메세지를 가로채거나 A의 비밀값에 대한 정보를 알아내려고 시도한다.

3.4. Replay attack

A와 B가 서버로부터 인증을 받은 올바른 사용자라고 가정하자. Replay attack은 이전의 A와 B사이의 정상적인 프로토콜에서 전송되었던 정보를 공격자가 가지고 있다가 나중에 A 또는 B에게 다시 보내서 프로토콜을 진행하려고 하는 공격이다. 이전에 사용되었던 정보를 다시 보내도 A와 B 사이에 정상적으로 수행되는 프로토콜과 비교해서 다를 것이 없다면 공격자는 결국 A 또는 B와 키를 교환할 가능성이 생긴다. 공격자와 키가 교환되면 Impersonation attack과 같이 공격자는 A 또는 B가 상대방에게 보내는 메세지를 가로챌 수 있다.

3.5. Known Key Security

공격자가 공격하고자 하는 세션 이전의 세션키를 알고 있지만 공격하고자 하는 세션의 키를 알 수 없는 경우 즉, 각 세션의 키가 독립적으로 생성되는 경우 프로토콜은 Known Key Security를 만족한다고 한다.

3.6. Perfect Forward Security

A가 서버로부터 인증을 받은 올바른 사용자라고 하자. 공격자가 A의 개인키나 패스워드를 알아냈다고 해도 이전에 A가 사용했던 어떠한 세션의 키도 알 수 없을 경우 프로토콜이 Perfect Forward Security를 만족한다고 한다. 본 논문에서 제안하는 프로토콜에서는 사용자들의 개인키 대신 패스워드와 스마트카드를 가지고 키를 교환하게 되므로 Perfect Forward Security에 대해 분석할 때, 패스워드를 알고 있고 스마트카드를 획득했을 경우에 대해 분석한다.

3.7. Unknown Key Share attack

서버로부터 인증을 받은 올바른 사용자인 A와 B의 프로토콜이 끝나고 각각 키를 계산했다고 가정하자. A가 자신이 계산한 키가 B가 아닌 다른 사용자와 공유되었다고 믿거나 B가 자신이 계산한 키가 A가 아닌 다른 사용자와 공유되었다고 생각하게 만드는 것을 Unknown Key Share attack이라고 한다.

3.8. Key Control

키 교환 프로토콜에서 키 교환을 하는 두 당사자 A와 B는 상호간에 설정될 키의 어떠한 부분도 키의 계산이

끝나기 전에 미리 결정지을 수 없어야 한다.

3.9. 부채널 공격

스마트카드에 대한 공격 방법으로 스마트카드 내부의 연산에서 일어나는 부가적인 요소인 소비전력이나 전자파 등으로 카드 내부의 저장된 값을 알아내는 방법이다. 본 논문에서 제안하는 프로토콜에서는 스마트카드를 사용하므로 안전성 분석에서 간단히 언급한다.

Ⅳ. 제안하는 프로토콜

본 장에서는 스마트카드를 이용해서 서버의 도움이나 공유된 비밀정보 없이 사용자간의 인증된 키 교환을 가능하게 하는 3자간 키 교환 프로토콜을 제안한다. 제안하는 프로토콜은 사용자가 서버에 등록하는 과정과 사용자간의 키 교환 과정으로 구성된다. 또한 추가적으로 패스워드 변경 과정이 있다. 사용자등록 과정은 사용자가 처음 스마트카드를 발급 받을 때 서버와 한번만 수행하고 스마트카드를 받은 사용자는 서버의 개입이 없이 다수의 사용자와 키 교환을 수행 할 수 있다. 제안하는 프로토콜의 세부적인 과정은 다음과 같다(그림 3).

4.1. 시스템 변수 설정

안전한 소수($p=2q+1$가 되는 소수 q가 존재하는 소수) p를 선택한 후, 위수가 q인 유한한 순환 곱셈군 G를 생성하고 G의 생성자 g를 선택한다. 그리고 임의의 랜덤한 길이의 문자열을 Z_q로 보내는 암호학적 해쉬함수 $H: \{0,1\}^* \rightarrow Z_q$를 생성한다. 또한 서버는 자신의 개인키 $x \in Z_q^*$를 선택하고 공개키 $y=g^x \pmod p$를 계산한다.

4.2. 사용자 등록

사용자는 서버에 등록을 원한다고 가정하자. 사용자가 등록을 신청하면 서버는 다음과 같은 단계로 사용자를 등록시켜주고 스마트카드를 발급한다.

(1) 사용자는 자신의 아이디(ID)와 패스워드(PW), 랜덤값(n)을 선택하고 해쉬키 HK, 트랩도어키 TK를 사용하는 트랩도어 해쉬함수 $h_{HK}()$를 생성해 $H(h_{HK}(PW,n))$를 계산한다.

(2) 사용자는 $H(h_{HK}(PW,n))$와 자신의 아이디(ID)를 안전한 경로로 서버에게 보낸다.

(3) 서버는 랜덤값 s를 선택해서 다음을 계산한다:

$$w = g^{H(h_{HK}(PW,n))} g^s \pmod p,$$
$$t = H(h_{HK}(PW,n)) + s + xH(ID,w) \pmod q.$$

(4) 서버는 w, t, g^s, y를 스마트카드에 입력하고 카드를 안전한 경로로 사용자에게 전달한다.

(5) 스마트카드를 받은 사용자는 자신이 처음 선택했던 랜덤값 n와 트렙도어 해쉬함수의 해쉬키 HK와 트렙도어 키 TK를 스마트카드에 입력한다.

4.3. 상호 인증 및 키 교환

사용자1과 사용자2가 안전한 통신을 위해 키를 교환하기 원한다고 가정하자. 키 교환은 다음과 같은 과정으로 이루어 진다.

(1) 사용자1은 자신의 스마트카드를 카드 리더기에 넣고 아이디 ID_1와 패스워드 PW_1'를 입력한다.

(2) 스마트카드는 입력된 패스워드 PW_1'으로 $w_1' = g^{H(h_{HK_1}(PW_1', n_1))} g^{s_1} \pmod p$ 를 계산한 후, 아이디의 확인과 카드에 저장된 w_1와 계산된 w_1'를 비교해 올바른 카드의 사용자인지 검증한다.

(3) 스마트카드는 랜덤값 r_1과 타임스템프 T_1로 다음 값을 계산한다.:

$$R_1 = g^{r_1} \pmod p,\ A_1 = r_1 + t_1 H(T_1, R_1) \pmod q.$$

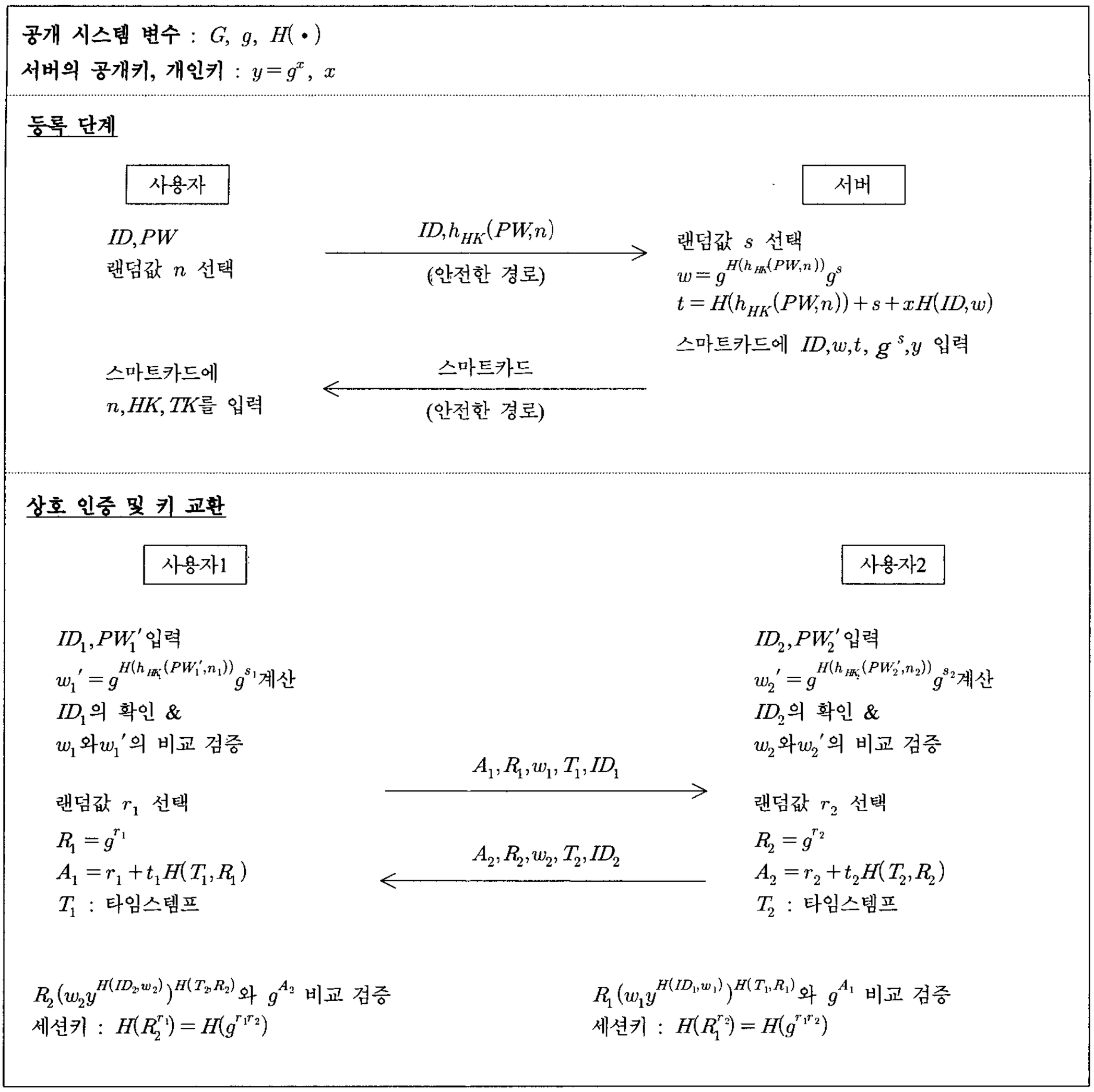
(4) 사용자1은 사용자2에게 A_1, R_1, w_1, T_1, ID_1를 전송한다. 사용자2는 같은 방법으로 A_2, R_2, w_2, T_2, ID_2를 생성해 사용자1에게 전송한다.

(5) 사용자1은 $R_2 (w_2 y^{H(ID_2, w_2)})^{H(T_2, R_2)} \pmod p$와 $g^{A_2} \pmod p$ 를 비교해서 사용자2를 인증하고 다음 값을 세션키로 계산한다:

$$key = H(R_2^{r_1}) = H(g^{r_1 r_2}).$$

4.4. 패스워드 변경

사용자1이 자신의 스마트카드의 패스워드를 변경하



〔그림 3〕 제안하는 키 교환 프로토콜

기 원한다고 가정하자. 사용자1은 다음의 단계를 수행해 자신의 패스워드를 교환하게 된다.

(1) 사용자1은 자신의 스마트카드를 카드 리더기에 넣고 아이디 ID_1와 패스워드 PW_1'를 입력한다.
(2) 스마트카드는 입력된 패스워드 PW_1'으로 $w_1' = g^{H(h_{HK_1}(PW_1', n_1))} g^{s_1} (\mathrm{mod} p)$를 계산한 후, 아이디의 확인과 카드에 저장된 w_1와 계산된 w_1'를 비교해 올바른 카드의 사용자인지 검증한다.
(3) 검증이 끝난 사용자는 자신이 사용할 새로운 패스워드 PW_1^*를 입력한다.
(4) 스마트카드는 트렙도어 키 TK_1를 사용해서 $h_{HK_1}(PW_1, n_1) = h_{HK_1}(PW_1^*, n_1^*)$이 되는 n_1^*를 찾아 n_1을 삭제하고 대신 저장한다.

이와 같이 변경된 카드는 패스워드 PW_1^*로서 검증이 가능해 진다. 하지만 w_1의 값은 변하지 않으므로 프로토콜 수행시에 카드내의 정보는 정상적으로 사용 가능하다.

V. 안전성 분석 및 효율성 비교

본 장에서는 본 논문에서 제안한 프로토콜에 대한 안전성을 분석한다. 3장에서 기술한 공격 방법과 안전성 개념들을 기준으로 제안한 프로토콜의 안전성에 대해 분석하고 안전함을 보인다. 또한 기존의 3자간 키 교환 프로토콜들과의 효율성을 비교한다.

5.1. Dictionary attack

On-line Dictionary attack은 패스워드 입력 횟수를 제안하는 등의 방법으로 간단히 막을 수 있으므로 Off-line Dictionary attack에 대한 안전성에 대해서만 분석한다. Dictionary attack을 하기 위해 필요한 정보는 상호인증 및 키 교환 단계에서 얻을 수 있다. 키 교환 과정의 (4)번 단계에서 공격자는 w_1를 얻을 수 있지만 공격자가 패스워드를 추측해 확인을 하려면 $h_{HK_1}(PW_1,n_1)$의 내부값인 n_1과 w_1을 생성하는데 사용한 s_1를 알아야 한다. 하지만 n_1, s_1는 랜덤한 값이기 때문에 공격자는 두 값을 알 수 없다. 따라서 제안하는 프로토콜은 Dictionary attack에 안전하다.

5.2. Insider attack

내부 공격자가 정상적인 프로토콜의 수행 중 상대방의 패스워드를 알아내려면 상호인증 및 키 교환 과정의 (4)번 단계에서 전송되는 w_1를 이용해서 패스워드를 추측하는 방법밖에 없다. 이것은 위의 Dictionary attack에서와 같은 이유로 불가능하다. 또한 상대방 스마트카드의 t_1값을 알아내는 것은 T_1에 대한 schnorr 서명[14]인 A_1, R_1을 가지고 비밀키를 알아내는 것과 같다. 따라서 t_1값을 알아내는 것 또한 불가능하다. 그러므로 내부 공격자가 다른 올바른 사용자의 어떠한 개인 비밀값도 알아내지 못한다. 이제 내부 공격자가 서버의 개인키를 알아내려고 한다고 가정하자. 서버의 개인키는 내부 공격자의 스마트카드에 있는 정보인 t_1를 이용해야 알 수가 있는데 스마트카드에 저장되어 있는 정보는 볼 수 없으므로 불가능하다. 공격자가 t_1를 알아낸다고 해도 s_1는 랜덤한 값이기 때문에 자신의 패스워드와 n_1, t_1으로서 서버의 개인키를 계산하는 것은 불가능하다.

내부 공격자는 일반적인 공격자와 다르게 올바른 자신의 패스워드와 스마트카드를 소지하고 있지만 위에서 살펴본 바와 같이 서버의 어떠한 정보도 알아낼 수 없고 또한 다른 사용자들의 비밀정보도 알아낼 수 없다. 또한 다른 올바른 사용자와의 프로토콜 수행시에도 공격자가 얻는 정보 이외의 어떠한 정보도 얻을 수 없다. 따라서 내부 공격자는 추가적으로 패스워드와 스마트카드를 소지하고 있지만 공격 능력은 일반적인 공격자와 다르지 않다.

5.3. Impersonation attack

사용자1을 가장하는 경우를 살펴보자. 공격자가 사용자1을 가장하여 사용자2와 정상적인 키 교환 프로토콜을 수행하기 위해서는 상대방에게 올바른 R_1과 A_1, w_1를 전송해야 한다. 하지만 서버에게 받아야하는 t_1없이 올바른 R_1과 A_1를 생성하는 것은 불가능하고 또한 w_1도 알 수 없고 따라서 올바른 값을 전송할 수 없다. 따라서 제안하는 프로토콜은 Impersonation attack에 안전하다.

5.4. Replay attack

제안된 프로토콜에서는 타임스템프 T_1를 사용하므로 이전의 통신 정보를 사용할 수 없어 Replay attack이 불가능하다. 공격자가 타임스템프만 바꿔서 올바른 정보를 생성해 전송하기 위해서는 공격 시점의 타임스템프에 대한 서명인 R_1과 A_1를 생성해야 한다. t_1없이 올바른 R_1, A_1의 생성은 불가능하므로 Replay attack에 안전하다.

5.5. Known Key Security

제안된 프로토콜의 교환된 키는 $H(g^{r_1r_2})$이다. 여기서 r_1, r_2는 키를 교환할 때마다 랜덤하게 선택되는 값이다. 따라서 공격자가 다른 어떠한 세션의 키를 알고 있다고 하더라도 공격을 하고자 하는 세션에서 사용되는 랜덤값 r_1, r_2와는 관련성이 없다. 따라서 내부 공격자의 공격에서 보았듯이 공격자가 키를 알아내려면 CDH 문제를 풀어야만 한다. 그러므로 제안하는 프로토콜은 Known Key Security를 만족한다.

5.6. Perfect Forward Security

공격자가 각 사용자의 패스워드를 알아내고 스마트카드를 획득했다고 해도 각 세션의 키를 생성할 경우에는 랜덤값 r_1, r_2를 사용하게 되므로 이전의 어떠한 세션에 대해서도 사용된 키의 정보를 얻을 수 없다. 따라서 이전의 세션에서 사용된 키를 알아내려면 공격자는 CDH문제를 풀어야 한다. 그러므로 제안된 프로토콜은 Perfect Forward Security를 만족한다.

5.7. Unknown Key Share attack

사용자1과 사용자2가 프로토콜을 수행할 때, 두 사용자는 각각 R_1, A_1와 R_2, A_2를 사용해 상대방에 대한 인증을 하게 된다. 이때 사용자2가 자신이 계산한 키가 사용자1이 아닌 사용자3과 설정된 키라고 믿으려면 사용자2가 키 교환 과정의 (4)번 단계에서 받는 정보가 사용자3의 아이디인 ID_3으로 검증되어야 한다. 하지만 공격자는 t_3를 알 수 없으므로 사용자3의 아이디인 ID_3으로 검증이 성공하는 값인 R_3, A_3를 생성할 수 없다. 마찬가지 이유로 사용자1도 자신과 키를 교환한 사람이 사용자2가 아닌 다른 사용자라고 믿을 가능성은 없다. 따라서 Unknown Key Share attack에 안전하다.

5.8. Key Control

제안된 프로토콜의 키는 두 사용자가 각각 랜덤값 r_1, r_2를 선택한 후, $H(g^{r_1 r_2})$를 계산해 키로 사용하게 된다. 프로토콜 중에 두 사용자는 한 번씩의 통신을 한 후 키를 계산하게 되고 또한 상대방이 선택한 랜덤값은 알지 못한다. 따라서 각 사용자는 설정될 키의 어떠한 부분도 미리 결정할 수 없다.

5.9. 부채널 공격

부채널 공격은 공격자가 스마트카드를 소지하고 스마트카드에 적당한 값을 입력한 후, 연산이 이루어지는 과정에서 나오는 부가적인 정보로 카드 내부에 저장된 값을 알아내는 방법이다. 본 논문에서 제안한 프로토콜의 스마트카드 내부에서 입력받은 값으로 처음 계산하는 것은 $g^{PW}(g^{TK})^n$인 트렙도어 해쉬함수의 계산이다. 따라서 입력받은 값으로 가장 처음 계산하게 되는 지수승을 부채널 공격에 안전하게 설계하면 그 이후의 연산은 안전하게 된다. 부채널 공격에 안전하게 지수승을 하는 방법에는 message blinding[14]과 exponent splitting[7] 등이 있다. 이러한 지수승을 제안한 프로토콜에서 사용하게 되면 부채널 공격을 방지할 수 있다.

5.10. 효율성 비교

제안한 프로토콜은 서버의 개입 없이 키 교환 과정을 수행 할 수 있다. 하지만 그로인해 사용자의 연산량이 증가하였다. 최근 Lu 등이 제안한 3자간 키 교환 프로토콜[19]에서 각 사용자는 3번의 지수승, 서버는 4번의 지수승이 필요하고 통신횟수는 3번이다. 본 논문에서 제안하는 사용자의 지수승이 트렙도어 해쉬함수의 계산까지 포함해 8번으로 증가하였지만, 서버가 계산하여야 할 지수승은 없고 통신횟수도 2번으로 줄였다. 사용자들의 연산량이 많이 증가하게 되었지만 최근 각종 PC와 같은 매체들의 연산 능력이 증가함에 따라 사용자의 연산량이 다소 증가하더라도 큰 무리가 없다. 따라서 통신횟수를 줄이는 것이 더 효율적인 프로토콜의 설계에 더 필요하다. 본 논문에서 제안한 프로토콜은 통신횟수가 2번으로 2명의 사용자가 상호 인증 및 키 교환을 하는데 필요한 최소한의 통신횟수이다. 하지만 키 교환 과정에 참여하는 서버의 경우는 키 교환을 하는 사용자가 증가함에 따라 지수적으로 증가(exponential increase)하게 된다. 따라서 서버의 연산 능력이 크다고 하더라도 서버의 연산량을 줄이는 것은 서버의 부담을 줄이기 위해 필요하다. 최근 제안되었던 스마트카드 기반의 3자간 키 교환 프로토콜[1,13]에서는 사용자와 서버의 연산을 비트연산과 해쉬함수로 처리해 지수승이나 곱셈보다 월등히 적은 연산량을 가지고 있다. 하지만 서버의 연산은 여전히 존재하고 통신량은 4번, 5번으로 본 논문에서 제안한 프로토콜보다 많다.

VI. 결 론

본 논문에서는 서버의 부담을 줄인 3자간 키 교환 프로토콜을 제안했다. 제안한 프로토콜은 기존의 3자간 키 교환 프로토콜과 마찬가지로 사용자들이 서로 다른 패스워드를 가지고도 키 교환이 가능하다. 또한 서버는 키 교환 과정에 개입을 하지 않아 연산 및 통신의 부담

이 줄어들고 사용자들의 패스워드를 저장할 필요도 없어져 저장의 부담도 줄어들게 되었다. 그리고 서버의 개입이 없이 키 교환을 하는 당사자끼리 통신을 하므로 서버의 수행을 기다릴 필요가 없어 다수의 사용자들이 키 교환을 수행하더라도 사용자들은 대기 시간 없이 즉시 키 교환이 가능하다. 점차적으로 회사와 같은 단체 내에서 스마트카드의 활용이 늘어나고 있다. 제안한 프로토콜은 스마트카드를 활용하고 있는 단체 내의 효율적인 통신과 관리를 위한 좋은 방법으로 사용할 수 있다.

참고문헌

[1] 전일수, "스마트카드를 이용한 3자 참여 인증된 키교환 프로토콜", 정보보호학회논문지, 제16권 제6호, pp.73-80, 2006.

[2] J.Baek, R.Safavi-Naini, W.Susilo, "Certificateless public key encryption without pairing", ISC 2005, Springer-Verlog, LNCS vol.3650, pp.134-148, 2005.

[3] SM.Bellovin, M.Merritt, "Encrypted key exchange: password-based protocols secure against dictionary attacks", Proceedings of the IEEE Symposium on Research in Research in Security and Privacy, pp.72-84, 1992.

[4] SM.Bellovin, M.Merritt, "Augmented encrypted key exchange: a password- based protocol secure against dictionary attacks and password file compromise", Technical report, AT&T Bell Laboratories, 1994.

[5] V.Boyko, P.MacKenzie, S.Patel, "Provably secure password authenticated key exchange using Diffie- Hellman", In B. Preneel, editor, Advances in Cryptology-Eurocrypt 2000, Springer-Verlag, LNCS vol.1807, pp. 156-171, 2000.

[6] H. Chung, W. Ku, "Three weaknesses in a simple three-party key exchange protocol", Information Sciences, vol.178, pp.220-229, 2008.

[7] C. Clavier, M. Joye, "Universal Exponentiation Algorithm, A First Step towards Provable SPA-Resistance", CHES 2001, Springer-Verlag, LNCS vol.2162, pp.300-308, 2001.

[8] M.Das, A.Saxena, V.Gulati, D.Phatak, "A novel remote user authentication scheme using bilinear pairings", Computer and Security, vol.25, no.3, pp.184-189, 2006.

[9] W.Diffie, M.E.Hellman, "New directions in cryptography", IEEE Trans., vol.IT-22, no.6, pp.644-654, 1976.

[10] L.Gong, M.Lomas, R.Needham, J.Saltzer, "Protecting poorly chosen secrets from guessing attacks", IEEE Journal on Selected Areas in Commu- nications, vol.11, no.5, pp.648-656, 1993.

[11] S.Halevi, H.Krawczyk, "Public-key cryptography and password protocols", ACM Transactions on Information and System Security, vol.2, no.3, pp.230-268, 1999.

[12] D.Jablon, "Strong password-only authenticated key exchange", ACM Computer Communication Review, vol.26, no.5, pp.5-26, 1996.

[13] W. Jaung, "Efficient Three-Party Key Exchange Using Smart Cards", Consumer Electronics, IEEE Transa- ctions on, vol.50, pp.619-624, 2004.

[14] P. Kocher, "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems", CRYPTO 96, Springer-Verlag, LNCS vol.1109, pp.104-113, 1996.

[15] H. Krawczyk, T. Rabin, "Chameleon Signatures", In Symposium on Network and Distributed Systems Security (NDSS'00), pp.143-154, 2000, Internet Society.

[16] T.Kwon, J.Song, "Secure agreement scheme for g^{xy} via password authentication", Electronics Letters, vol.35, no.11, pp.892-893, 1999.

[17] L. Law, A. Menezes, M. Qu, J. Solinas, S. Vanstone, "An Efficient Protocol for Authenticated Key Agreement", Designs Codes and Cryptography, vol.28, pp.119-134, 2003.

[18] T. Lee, T. Hwang, C. Lin, "Enhanced three-party encrypted key exchange without server public keys", Computer & Security, vol.23,

pp.571-577, 2004.

[19] R. Lu, Z. Cao, "Simple three-party key exchange protocol", Computers & Security, vol.26, no.1, pp.94-97, 2007.

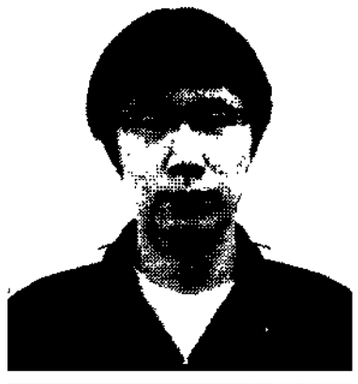
[20] N. McCullagh, P.S.L.M. Barreto, "A New Two-Party Identity-Based Authe- nticated Key Agreement", CT-RSA 2005, Springer-Verlag, LNCS vol.3376, pp.262-274, 2005.

[21] C.Schnorr, "Efficient identification and signatures for smart cards", Crypto'89, pp.239-252, 1990.

[22] A.Shamir, Y.Tauman, "Improved Online/Offline Signature Schemes", CRYPTO 2001, Springer-Verlag, LNCS vol.2139, pp.355-367, 2001.

[23] E.Yoon, W.Lee, K.Yoo, "Secure remote user authentication scheme using bilinear pairings", WISTP 2007, Springer-Verlag, LNCS vol.4462, pp.102-114, 2007.

〈著者紹介〉



김 용 훈 (Yong-Hun Kim) 학생회원

2006년 8월: 광운대학교 수학과 졸업
2006년 9월~현재: 고려대학교 정보경영공학전문대학원 정보보호학과 석사과정
<관심분야> 암호 이론, 정보보호 이론, 암호 프로토콜

윤 택 영 (Taek Young Youn) 학생회원

2003년 2월: 고려대학교 수학과 이학학사
2005년 2월: 고려대학교 정보경영공학전문대학원 정보보호학과 공학석사
2005년 3월~현재: 고려대학교 정보경영공학전문대학원 정보보호학과 박사과정
<관심분야> 암호 이론, 정보보호 이론, 암호 프로토콜, 부채널 공격

박 영 호 (Young Ho Park) 정회원

1990년 2월: 고려대학교 수학과 이학사
1993년 2월: 고려대학교 수학과 이학석사
1997년 2월: 고려대학교 수학과 이학박사
2002년 3월~현재: 세종 사이버 대학교 부교수
<관심분야> 정수론, 공개키 암호, 암호 프로토콜, 부채널 공격