그룹 환경의 사용자 인증 및 키 교환 서비스 프로토콜 연구

변진욱* · 이수미** · 이동훈***

A Study on the User Authentication and Key Exchange Service for Group Environment

Jin-Wook Byun* · Su-Mi Lee** · Dong-Hoon Lee***

■ Abstract ■

Over the years a password has been used as a popular authentication method between a client and a server because of its easy-to-memorize property. But, most password-based authentication services have focused on a same password authentication scheme which provides an authentication and key exchange between a client and a server with the same password. With rapid change of communication environments in the fields such as mobile networks, home networking, etc., the end-to-end security allowing users to hold different password is considered as one of main concerns.

In this paper, we consider a new authentication service of how each client with different own password is able to authenticate each other, which is a quite new service paradigm among the existing services. This new service can be used in the current or next generation network environment where a mobile user in cell A wants to establish a secure end-to-end channel with users in cell B, C, and D using only their memorable passwords. This end-to-end security service minimizes the interferences from the operator controlled by network components. To achieve this end-to-end security, we propose an authentication and key exchange service for group users in different realm, and analyze its security in a formal way. We also discuss a generic construction with the existing authentication schemes.

Keyword : Authentication, Different Password Authentication, Group Service

논문투고일 : 2009년 02월 09일 논문수정완료일 : 2009년 06월 04일 논문게재확정일 : 2009년 06월 09일

* 평택대학교 정보통신학과
** 금융보안연구원
*** 고려대학교 정보경영공학전문대학원

1. 서 론

유비쿼터스 컴퓨팅 기술은 초고속 인터넷, 모바일, 디지털 컨버전스 등으로 대표되는 차세대 이동통신 기술의 급속한 확산을 불러 오고 있으며 이와 더불어 사이버 윤리 문제와 같은 법과 제도의 변화를 야기시키고 있다. [1]에서 언급되었듯이 유비쿼터스 시대의 차세대 이동통신 기술은 다양한 시나리오에서 복잡한 요구사항들을 포함하고 있으므로 이를 위해 안전하고 효율적인 차세대 IT 서비스 기술이 필요하다. 차세대 유비쿼터스 서비스 사회에서 공통적으로 적용 되는 기술로는 초고속 데이터 전달 기술, 상호 연동 및 컨버전스 기술, 가상현실 기술, 상황인지 기술, 서비스 연속성 기술, 보안 및 프라이버시 기술들이다[1]. 구체적으로 서비스를 전달 받는 측면에서는 고속 데이터의 전달 기술, 컨버전스 기술 등이 중요하겠지만, 이와 더불어 서비스에 대한 안전성, 개인정보보호 기술 및 인증 문제는 데이터의 전달 관련 기술 못지않게 대단히 중요하다. 예를 들어 정보보호 서비스가 보장되지 않은 컴퓨팅 서비스의 제공은 실효성을 확보하기 힘들다. 더욱이 최근 각종 금융망 해킹 사건, 웜 확산 및 프라이버시 노출 사건(연예인 x-file) 들로 인해서 정보보호 서비스는 서비스 컴퓨팅에서 반드시 구현되어야 할 필수 사항으로 쟁점화 되었다.

차세대 IT 컴퓨팅 환경에서 반드시 보장되어야 할 정보보호 서비스에는 무결성, 인증, 부인봉쇄, 기밀성이 있다. 그 중에 인증 서비스는 서버가 제공하는 다양한 서비스에 대한 접근성을 부여하므로 더욱 중요한 정보보호 서비스이다. 그러므로 인증 서비 기술은 기존의 안전한 암호, 해쉬함수와 같은 암호학적 프리미티브(primitive)들을 이용하여 주의 깊게 설계된다.

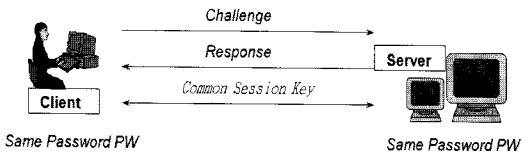
인증 서비스 중 가장 널리 쓰이고 있는 기술은 패스워드 인증 기술이다. 패스워드는 사용자가 암기하기 편리하다는 효율성으로 인해 현재 IT 환경 및 차세대 클라이언트-서버 인증 서비스에 변함없이 널리 사용될 것으로 확신한다. 하지만 암기하기 편리하다는 사실은 패스워드가 선택되어지는 공간이 계산적으로 협소함을 의미하므로 공격자에게 사전공격(dictionary attack)을 용이하게 하는 단점을 지닌다. 이러한 사전공격의 취약점을 해결하고 패스워드 인증에 필요한 계산량 및 통신량을 줄인 효율적인 패스워드 인증 방법이 기존에 많이 연구되었다[1, 5, 18, 24]. 최근에는 통합서비스 차원에서 사용자 인증, 메시지 기밀성, 무결성 서비스들을 동시에 제공하는 프로토콜이 업계에 많은 관심을 받게 되었다.

메시지의 기밀성, 무결성 및 인증 서비스를 제공하기 위해서는 사전의 키 교환 프로토콜이 선행되어야 한다. 키 교환 프로토콜은 안전하지 않은 분산 네트워크 환경 하에서 비밀 값을 가지고 있는 두 사용자 혹은 그 이상의 사용자들이 공통의 비밀 키 값을 동의할 수 있도록 해 준다. 생성된 비밀키 값들은 메시지의 기밀성을 제공하기 위한 대칭키 암호 알고리즘(예 : 3DES)의 암/복호화키로 사용된다. 또한, 메시지의 무결성 및 인증(MAC : message authentication code) 을 위한 대칭키로 사용된다. 이러한 키 교환 프로토콜은 사용자들 간의 안전한 통신을 구축하는데 필수적이므로 키 교환 기술은 IPsec, SSL등과 같은 정보보호 서비스 프로토콜에 핵심 기술로 널리 사용되고 있다. 특히 사용자의 암기 가능한 패스워드를 이용하여 인증 및 키 교환을 동시에 제공하는 패스워드 인증 키 교환(PAKE : password authenticated key exchange) 기술이 사용자 인증 및 비밀 키 값 공유를 위한 실용적인 기술로 많은 관심을 받고 있으며, 사전 공격에 안전하고 효율적인 PAKE 프로토콜들이 각각 다른 암호학적 가정들을 기반으로 하여 제안되었다.

1.1 관련 연구

문헌에서 언급되어지는 대부분의 PAKE 프로토콜은 사용자와 서버간의 사전 공유된 동일한 패스

워드에 관한 서비스에 초점이 맞추어져 있다. [그림 1]과 같이, 사용자가 먼저 서버에게 패스워드를 안전하게 등록한 후에, 시도-응답(challenge-response) 방법으로 인증을 수행한다. 사용자와 서버는 공유된 패스워드를 이용하여 시도-응답 방법으로 공통의 세션 키를 만들고 형성된 세션 키에 대한 키 확인 과정을 수행한다. 논문에서 언급되는 대부분의 PAKE 프로토콜은 이러한 SPA(shared password authentication) 모델에 초점이 맞추어져 있다[18, 13, 15, 23, 24]. 즉, SPA 모델은 사용자와 서버간의 동일한 패스워드에 관한 인증 및 키 교환을 제공한다.



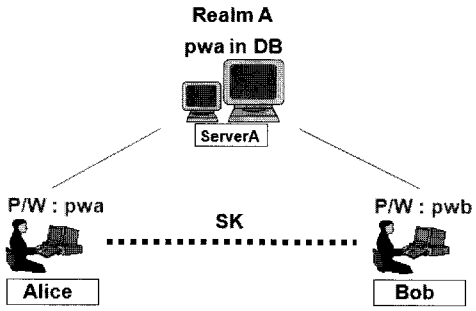
[그림 1] SPA 모델의 패스워드 인증 방법

Bellovin과 Merrit는 SPA 모델에서 처음으로 사전공격에 안전한 EKE 스킴을 제안하였는데, 이는 SPA 모델 이후 많은 연구결과물의 기초가 되고 있다[18]. 최근, Bresson 등은 사용자들이 동일한 패스워드를 공유하고 있는 상황에서 패스워드 인증 그룹 Diffie-Hellman 키 교환 프로토콜을 제안하였다[12].

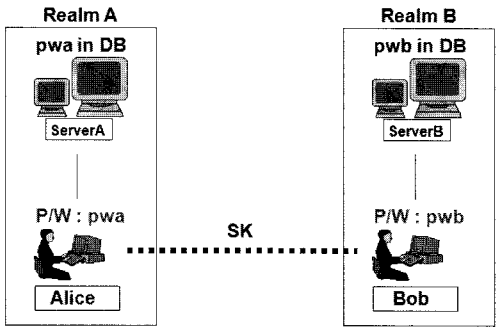
모바일 네트워크와 Ad-hoc 네트워크와 같은 통신 환경의 빠른 변화에 의해 사용자간의 종단간 인증은 중요한 관심사로 대두되고 있다. 최근 보다 효율적인 종단간 인증을 위해 두 사용자들의 서로 다른 패스워드를 이용해서 인증 된 키 교환을 하는 방식들이 제안되어졌다[1, 20, 29]. 이러한 DPA(different password authentication) 모델에서 사용자들은 자신의 서로 다른 패스워드를 이용하여 서버의 도움을 받아 공통의 세션 키를 만들어 낸다. 이러한 DPA 모델은 보다 적은 메모리 공간을 요하는 모바일 환경에서 사용자들 간의 안전한 종단 간 인증이 필요한 환경에 적합하게 사용된다.

DPA 모델은 [그림 2]와 [그림 3]에 나타내었듯이, 사용자의 영역이 동일한 단일 영역과 상이한 다중 영역의 DPA 환경으로 구분된다. 전자는 인증과 키 교환을 원하는 사용자가 동일한 서버에 등록되어 있는 경우를 의미하고, 후자는 각각 다른 서버에 등록되어 있는 환경을 말한다. 일반적으로 두 사용자를 선택했을 때 각자 고유의 서버를 가지는 경우가 많으므로, 후자가 더 일반적인 통신 모델이라 할 수 있다.

이러한 DPA 문제를 해결하기 위해, Steiner 등은 처음으로 단일 서버를 이용해 서로 다른 패스워드를 가진 두 사용자간에 공통의 세션 키를 형성 해주는 3-Party EKE 프로토콜을 제안하였다[29]. 최근에는 변진욱 등이 단일 영역 환경 및 다중 영역 환경에서 C2C-PAKE(client to client password-authenticated key exchange) 프로토콜을 제안하였으며 이에 대한 안전성도 분석하였다.



[그림 2] 단일 영역 간 DPA 환경



[그림 3] 다중 영역 간 DPA 환경

1.2 논문의 공헌도 및 구성

사용자들이 인증 및 키 교환을 수행하고자 할 때 서로 다른 패스워드를 가지고 있는 경우가 대부분이다. 그러므로 DPA 모델이 SPA 모델보다 더욱 활용도가 높은 인증 모델이다. 더욱이 차세대 유비쿼터스(ubiquitous) 환경에는 언제 어디서나 인증 서비스가 이루어져야 하므로 DPA 모델이 보다 활용성이 높은 인증 서비스로 주목 받을 것이다. 하지만, 지금까지의 DPA 관련 연구는 두 사용자들에 대해서 국한되었다. 즉, DPA 환경의 그룹 환경(통신 사용자가 n명인 경우)에 대한 연구가 아직 수행되지 않았다. 예를 들어, 통신 참여자가 3명인($n=3$) 경우를 고려해 보자. 즉, 사용자 A, B, C가 다음(daum) 서버, 네이버(naver) 서버 및 구글(google) 서버[1]에 각각 다른 패스워드를 가지고 등록된 사용자라 가정했을 때, 이들 사용자들을 위한 인증 및 키 교환 프로토콜 서비스가 필요하다. 이는 사용자가 오직 2명인 기존의 연구보다 [1] 더욱 일반화된 인증 서비스이며, 차세대 분산 네트워크 인증 서비스에 널리 활용될 것이다. 이러한 그룹 환경의 인증은 그룹 환경에 맞는 안전성 모델이 필요하기 때문에 효율성과 안전성을 지니면서 확장시키는 것이 결코 쉬운 작업이 아니다. 보안 프로토콜의 설계는 안전성 분석을 위한 안전성 모델 구축을 필요로 하므로 설계와 함께 보안적인 요소가 함께 고려되어져야 한다.

본 논문에서는 다중 영역환경에서 그룹 사용자들이 서로 다른 패스워드를 가지고 상호 인증 및 키 교환이 가능한 프로토콜을 제안한다. 이에 적합한 안전성 모델을 설립하고 제안된 프로토콜에 대한 안전성을 분석한다.

이를 위해 현재까지 제안되었던 인증된 키 교환 프로토콜(AKE : authenticated key exchange)을 비밀 키 기반, 패스워드 기반, 하이브리드 기반으로 분류하여 살펴보면서 현재 연구 이슈 및 동향, 향후 연구 과제 등에 대해서 분석한다. 그 후, 다중 영역 환경에서의 그룹 사용자들을 위한 상호 인증 및 키 교환 프로토콜을 제안하고 제안된 프로토콜이 안전함을 여러 암호학적 가정을 이용하여 증명한다. 또한 포괄적인(generic) 구축 방법론과 그 안전성에 대해서도 살펴본다.

1) http://www.daum.net, http://www.naver.com, http://www.google.com.

2. 인증된 키 교환 프로토콜 분류 및 동향

인증된 키 교환 프로토콜(AKE : authenticated key exchange)은 인증과 키 교환이 동시에 이루어지는 프로토콜을 의미한다. 이 장에서는 현재까지 제안된 AKE 프로토콜의 연구 결과를 살펴본다. 인증 및 키 교환을 원하는 사용자들은 개인의 비밀 정보(secret information, authentication information)가 반드시 필요하다. 이 비밀 정보의 형태에 따라 비밀 키 기반 키 관리 기법, 패스워드 기반 키 관리 기법, 하이브리드 키 관리 기법으로 나눈다. 하이브리드 키 관리 기법은 패스워드와 비밀 키 관리 기법을 병합해서 사용하는 인증된 키 교환 프로토콜이다.

2.1 비밀 키 기반 키 관리 기법

비밀 키 기반 키 관리 기법은 PKI(public key infrastructure)기법을 전혀 사용하지 않는 인증된 키 교환 기법이다. 즉, 오직 대칭키 기반의 암호 기술(예 : 3DES, AES)들을 이용하므로 빠른 속도를 보장받을 수 있다. 이는 다시 대칭적인(symmetric) 기술과 비대칭적인(asymmetric) 기술로 나뉜다. 이에 대한 설명은 다음과 같다.

- 대칭적인 기술 : 대칭적인 기술은 개체들 간에 미리 공유된 대칭키를 이용해서 공통의 세션 키를 만들어 내는 방법이다.
- 비대칭적인 기술 : 비대칭적인 기술은 개체들 간에 서로 다른 키를 이용하여 공통의 세션

키를 만들어 내는 방법이다. 서로 다른 키를 이용하므로 개체들 간에 서버의 개입이 필수적이다. 또한 Diffie-Hellman과 같은 공개키 기술들이 포함된다. 하지만 공개키, 개인키, 인증서와 같은 PKI 기술들을 요구하는 것이 아니라, 프로토콜에서 Diffie-Hellman 기술에 필요한 공통의 공개키 파라메터(예를 들어, 생성자 g, 나머지 연산자 p) 값만 사용되므로 기존의 PKI 기술들보다는 구현 및 계산 비용 측면에서 훨씬 효율적이다. 참고문헌 [4]에서도 공통의 Diffie-Hellman 공개키 파라메터를 이용하여 ad-hoc 기반의 키 관리 기법을 제시하였다. 그러므로 비대칭적인 기술은 공개키 파라메터를 이용하여 효율적으로 설계한 프로토콜을 포함한다.

2.1.1 대칭적인 기술 기반

대칭적인 기술은 인증된 키 교환 프로토콜에서 대표적이고 고전적인 기술이다. 많은 프로토콜들이 제안되어졌으며, 아직까지도 센서(sensor) 네트워크 및 제약된 환경에서 키 교환을 위한 모델로써 많이 활용되고 있다. 1993년에 처음으로 Bellare 등이 대칭적인 기술 기반에 대한 안전성 모델(security model)을 구축하였으며, 이를 기반으로 처음으로 제안된 프로토콜의 안전성을 수학적으로 증명하였다[17]. 1995년에 이들은 이 프로토콜을 개체가 3인 환경으로 확장하였다. 즉, 서버가 중간에 있고 두 개체가 서버와 동일한 비밀 키를 사전에 공유하고 있는 환경에서 두 개체 간에 안전한 인증 및 키를 만드는 프로토콜을 제안하였다.[2] 이 분야의 연구 이슈는 개체가 n인 다자간 환경에서의 동적 구성원에 대한 효율적인 프로토콜 설계이다. 즉, 다자간 환경이므로 구성원들이 입/출입할 때에 발생하는 네트워크 오버헤드를 줄이면서 구성원들의 인증 및 키 교환을 이끌어 내는 것이 최근 주된 연구주제이다.

2.1.2 비 대칭적인 기술 기반

비 대칭적인 기반 인증 및 키 교환 기법은 PKI 기술을 이용하며, Diffie-Hellman 기반의 어려운 수학적 가정들에 기반하여 설계되어졌다. 먼저 양자간(two party)에서 프로토콜은 MTI, KEA, Unified Model 등의 프로토콜들이 제안되어졌다[7]. 3라운드의 통신 라운드수를 가지며, 안전성 증명도 Bellare-Rogaway 모델에 의해 수학적으로 이루어졌다. 다자간 환경에서는 Bresson 등이 2001년에 양자간 Diffie-Hellman 키 교환 기법을 다자간으로 확장하여 인증된 그룹 키 기법을 제시하였다[11]. 또한 동일한 해에 위의 프로토콜을 동적인 환경으로 확장하여 제시하였으며[14], 2002년도에는 표준적인 모델(standard assumption)에 기반한 그룹 키 관리 기법을 동적인 환경에서 제안하였다[13]. 2003년도에 Katz와 Yung은 상수 라운드를 가지는 그룹 키 교환 기법을 처음으로 제시하였으며, 그 다음해에 김현정 등은 상수 라운드를 가지는 그룹 키 교환 기법을 동적인 환경에서 설계하였다[9]. 2004년에는 정익래 등이 다자간 환경에서 상수 라운드를 요하는 프로토콜들을 표준적인 모델에서 제안하였다[10].

ID 기반 키 교환 프로토콜도 비대칭적인 기술에 포함된다. ID 기반 암호는 사용자 이름, IP 주소, 전화번호, 이메일 주소로부터 사용자의 공개키를 추출하여 사용함으로써, 기존 공개키 암호의 인증서 관리의 문제를 효과적으로 해결한 암호 기법이다[19]. 이 암호 기법에 기반하여 제안된 것이 ID 기반 키 교환 프로토콜이다. ID 기반 키 교환 프로토콜은 사용자의 ID로부터 직접 공개키를 추출할 수 있다는 편리성으로 인해 최근 급격히 이슈화되는 분야이다. 2005년에 McCullagh 등은 새로운 ID 기반 양자간 키 교환 프로토콜을 제안하였다. 실제로 2004년에 최규영 등은 개체가 3인 환경에서 상수 라운드를 요하는 프로토콜을 제안하였다. 최

2) 두 개체가 서로 다른 키를 가지고 있으므로 비대칭적인 기술이라 말할 수도 있겠지만, 서버와 개체들 간에는 동일한 키를 이용하므로 대칭적인 기술로 분류하였다.

근 표준적인 모델에서 상수 라운드를 요하는 ID 기반 키 교환 프로토콜을 설계하는 것이 주요 이슈이다. 이와 더불어, 동적인 환경에서 ID 기반 키 교환 프로토콜 역시 아직 설계되지 않았다. 이에 대한 향후 연구가 필요하다.

2.2 패스워드 키 기반 키 관리 기법

패스워드 역시 대칭적인 기술과 비대칭적인 기술로 나뉜다. 대칭적인 기술은 개체들 간에 공통의 패스워드를 미리 공유해서 공통의 세션 키를 만들어내는 기술이다. 논문의 서론에서 설명한 SPA 모델을 의미한다. 비대칭적인 기술은 개체들 간에 서로 다른 패스워드를 미리 공유해서 공통의 세션 키를 만들어 내는 기술이며, DPA 모델을 의미한다. 위급한 상황의 ad-hoc 환경(전쟁이나 긴급사고) 에서는 개체들이 동일한 패스워드를 소유하고 있다는 가정이 비현실적일 수 있다. 그러므로 비록 서버의 개입이 반드시 필요하지만, 서로 다른 패스워드를 이용한다는 측면만 놓고 비교 했을 때, 비대칭적인 기술이 ad-hoc 네트워크 및 급변하는 차세대 네트워크 환경에 좀 더 현실적인 기술이라 할 수 있다.

2.2.1 대칭적인 기술 기반(SPA 모델)

1992년도 Bellovin과 Merrit 가 처음으로 패스워드 기반 키 교환 기법인 EKE 프로토콜을 제시하였다[7]. 이 결과는 많은 패스워드 기반 키 교환 프로토콜들의 설계상의 핵심 원리로 적용되어지고 있다. 2000년도에 Bellare 등은 EKE 프로토콜의 안전성 모델을 세웠으며, 정의된 모델을 바탕으로 EKE 프로토콜이 안전함을 계산적 Diffie-Hellman 가정을 기반으로 하여 수학적으로 보였다[15]. 2001년도에 Bresson 등은 EKE 프로토콜을 다자간 환경으로 확장하였다[12]. 하지만 참고문헌[12]의 프로토콜은 요구되는 라운드 수가 사용자의 수에 의존적이어서 확장적(scalable)이지 못하며 비효율적이다. 또한 이상적인 암호(ideal cipher[3])를 가정하고 있어서 표준적인 가정으로 안전성을 증명해야 할 여지는 여전히 남아 있다.

2.2.2 비 대칭적인 기술 기반(DPA 모델)

비 대칭적인 기술은 최근 유비쿼터스 환경 및 분산환경 기술의 발전으로 인해 많은 관심을 받고 있는 분야이다. 1994년, Steiner 등은 처음으로 단일 서버를 이용해 서로 다른 패스워드를 가진 두 사용자간에 공통의 세션 키를 형성하는 3-Party EKE 프로토콜을 제안하였다[29]. 하지만 Ding과 Horster는 3-Party EKE 프로토콜이 감지되지 않는 온라인 사전공격(undetectable online dictionary attack)에 취약함을 보였다[22]. 또한 Lin등은 3-Party EKE 프로토콜이 오프라인 사전공격에 취약함을 보였으며, 그들은 서버의 공개키를 사용하여 위의 오프라인 및 감지되지 않은 사전공격에 강한 LSH-3PEKE 프로토콜을 제안하였다[26]. 하지만 LSH-3PEKE 프로토콜은 사용자들이 서버의 공개키를 얻어서, 확인하는 추가적인 작업을 수반한다. 이에 Lin 등은 서버의 공개키를 필요로 하지 않고 위의 두 공격에 강한 새로운 LSSH- 3PEKE 프로토콜을 제안하였다[27]. 최근에는 변진욱 등이 다중 영역 환경과 단일 영역 환경에서의 C2C-PAKE (client to client password-authenticated key exchange) 프로토콜을 제안하였으며, 이에 대한 안전성도 비형식적으로(informally) 분석하였다 [1, 20]. 참고문헌[20] 프로토콜은 이후 새롭게 재설계 되어졌다[28, 25, 30]. 먼저, [30, 25] 논문에서[20] 프로토콜이 다중 영역에서 안전하지 않음을 보였고, [28]에서는 [25]에서 새롭게 제안된 프로토콜이 UKS (unknonw key share) 공격에 안전하지 않음을 보였다. 또한 UKS 공격에 안전한 프로토콜을 재설계하였다. 이 분야에서는 공학적인 접근과 더불어

3) 입력과 출력의 길이가 동일한 랜덤 일대일 permutation 함수(random one-to-one permutation)이다.

4) 대칭적인 기술 기반에서는 개체들이 공통의 비밀 값을 저장하고 있기 때문에 개체들에게 입/출입을 허용했을 때 그 안전성을 절대 보장 받을 수 없다. 왜

〈표 1〉 인증된 키 교환 기법 연구 결과 정리

분 류	형 태	참여자		Round	Dynamic	Provable security	Security model
비밀키 기반 Secret key	Symmetric	2-party		3	-	Yes	Standard
		3-party(S)		5	-	Yes	Standard
		N-party(S)		N/A	Yes	?	
	Asymmetric	2-party		3	-	Yes	Standard
		N-party(S)		3	Yes	Yes	Standard
		N-party		3	Yes	Yes	Ideal Hash
	Asymmetric (ID-based)	2-party		3	-	Yes	Ideal Hash
		N-party		3	No	Yes	Ideal Hash
패스워드 기반 Password	Symmetric	2-party		3	-	Yes	Standard
		N-party		3	No[4]	Yes	Ideal Hash
	Asymmetric	3-party		6	-	Yes	Standard
		N-party	단일	3	No	Yes	Ideal Hash
			다중	연구되지 않았음			
하이브리드기반 Secret key +Password	Symmetric	2-party		연구되지 않았음			
		N-party					
	Asymmetric	2-party		3	-	Yes	Standard
		N-party		3	-	Yes	Standard

주) 2-party : 양자간, 3-party : 삼자간, n-party : 다자간, provable security : 안전성 증명 여부, Security Model : 안전성 증명 모델, - : 제안된 프로토콜이 없음.

이론적인 접근도 많이 이루어졌다. 2004년에 Abdalla 등은 기존의 양 자간 패스워드 기반 키 교환 프로토콜을 기반으로 하여 삼자간 환경에서 키 교환 프로토콜을 구성하는 일반적인 구성법에 관한 방식을 소개하였고[2], 2005년에는 이러한 일반적인 구성법을 효율적으로 구성하였다[3]. 2004년에 다자간 환경에서의 서로다른 패스워드를 이용한 키 교환 기법들이 제안되어졌다. 총 2개의 기법들이 제안되어졌는데, 하나는 유니캐스트 모델에서 제안되어졌고(N-party EKE-U 프로토콜), 다른 하나는 멀티캐스트 모델에서(N-party EKE-M 프로토콜) 제안되어졌다[6].

비대칭적인 기술 기반에서는 크게 두 가지의 해결과제가 남아 있다. 첫째, 여러 사용자들이 참여하는 그룹 환경에서의 상호 인증 및 키 교환 서비스를 제공하는 프로토콜 설계이다. 둘째는 표준적인 모델에서 제안된 프로토콜의 안전성을 증명하는 것이다.

2.3. 하이브리드 키 기반 키 관리 기법

하이브리드 기법 역시 크게 대칭적인 기반과 비대칭적인 기반 기술로 분류된다. 대칭적인 기반 기술은 패스워드 기법과 비밀 키 기법을 병행해서

냐하면, 구성원 모두가 동일한 비밀 값을 가지고 있기 때문이다. 그러므로 대칭적인 기술 기반에서는 동적인 환경을 고려하지 않는다.

사용하는 방법이고, 비 대칭적인 기술은 패스워드 기법과 비밀 키 기법과 패스워드 기법을 병행해서 사용한다.

2.3.1 대칭적인 기술 기반

아직까지 비 대칭적인 기술이 주를 이루고 있으며, 대칭적인 기술은 아직 연구된 바 없다. 그 이유로는 사용되는 응용 분야를 찾기 힘들다는데 있다. 비밀 키 기반 기술을 가지고 충분히 세션 키를 만들 수 있으며, 또한 패스워드만을 가지고도 세션 키를 만들 수 있다. 이 두 가지 장점들을 적절히 적용할 수 있는 응용분야를 찾는 것은 쉽지 않다. 이와 반면에 비 대칭적인 기술은 비 균형적인 네트워크에 적절히 사용될 수 있다.

2.3.2 비 대칭적인 기술 기반

대칭적인 기반 기술은 공통의 세션 키를 만들기 위해 개체들이 패스워드 혹은 비밀 키를 이용한다. 예를 들어, 계산 및 자원에 제약이 많은 개체와 이와 반대로 우수한 자원을 가지고 있는 개체들 간에 통신을 하기를 원한다고 가정하자. 제약된 자원을 가진 사용자는 패스워드만 암기하고, 우수한 자원을 가진 개체는 PKI 기술을 이용할 수 있다. 이러한 환경이 하이브리드가 사용될 수 있는 대표적인 분야이다. 즉, 하이브리드 방법은 비균형적인(unbalanced) 네트워크에서 사용이 될 수 있다. 1999년에 Halevi 와 Krawczyk는 처음으로 이러한 환경에서 키 교환 프로토콜을 설계하였으며, 그 안전성을 증명하였다[24]. 또한 2000년도에는 Boyarsky가 다중 사용자 환경으로 위의 프로토콜을 확장하였다[5]. 비 균형적인 네트워크의 일반적인 예로는 모바일 통신 환경이 될 수 있다. 즉, 모바일을 가진 사용자가 자원이 상대적으로 풍부한 AP와 통신하는 환경이다[24]. 프로토콜보다 계산적인 측면에서 더 효율적인 프로토콜을 모바일 환경에 맞게 설계하는 것이 최근 이슈이다. 지금까지 논의했던 인증된 키 교환 프로토콜에 관한 연구결과를 <표 1>에 나타내었다.

3. 패스워드 기반 그룹 환경의 안전성 모델

본 장에서는 서로 다른 패스워드를 이용한 그룹 환경에서의 인증 및 키 교환 프로토콜 안전성을 정의한다. 안전성을 정의하기에 앞서, 프로토콜 참여자, 공격자의 능력, 구성 알고리즘을 정의한다. 안전성 모델은 참고문헌[15]에 정의된 안전성 증명과 모델을 바탕으로 하고 있다. [15]의 모델은 n명의 사용자가 동일한 패스워드를 이용하여 세션 키를 형성하도록 설계되었다. 이를 n-1개의 서로 다른 패스워드 환경에 맞게 변형한다.

3.1 안전성 모델

3.1.1 참여자

두 종류의 프로토콜 참여자가 존재한다. 사용자와(Clients) 서버(Server)이다. ID = $Clients \cup Server$라 했을 때, ID는 동적으로 변하지 않고 항상 고정되어 있다고 가정한다. 서버 S 는 단일 서버로 구성되어 있고, $Clients = \{C_1, \cdots, C_{n-1}\}$라 정의한다. 각 사용자 C_i 비밀 패스워드 pwi를 소유하고 있으며 S는 자신의 데이터베이스에 패스워드정보를 보관하고 있다. 사용자 C_i는 키 교환 프로토콜을 여러 번 실행할 수 있으며, C_i에 의해 수행되는 프로토콜의 t번째 인스턴스를 Π_{ti}로 표기한다.

3.1.2 알고리즘

제안하는 프로토콜은 다음 세 개의 주요 알고리즘으로 구성되어 있다.

- 패스워드 생성 알고리즘 GPW : 입력으로 1^k를 받아서 사용자 C들의 패스워드 pw를 출력한다. 단 k는 안전성 파라미터이다.
- 사용자 등록 알고리즘 R^U : 고정된 사용자를 입력으로 받아서 각각의 사용자들에 해당하는 패스워드들을 서버 S에 등록한다.
- 세션 키 형성 알고리즘 GSK : 고정된 사용자

들을 입력으로 받아서 서버의 도움을 받아서 사용자들의 다른 패스워드를 이용하여 공통의 세션 키를 만들어 낸다.

본 논문에서는 네트워크 자원을 이용하여 프로토콜 구성원들의 대화를 조정할 수 있는 능동적인 공격자가 있음을 가정한다. A는 임의의 Π_{ti}에 언제든지 접근할 수 있으며, 허용되는 질의의 내용을 요약하면 다음과 같다.

- Send(Π_{ti}, M) : 이는 오라클 Π_{ti}에게 메시지 M을 전달한다. Send 질의는 공격자 A가 프로토콜 내에서 사용자 C_i에게 메시지를 전달하는 행동을 모델화 한 것이다.
- Reveal(Π_{ti}) : 이는 오라클 Π_{ti}가 자신의 파트너와 함께 세션 키를 계산하였을 경우, 허용되는 질의로써, 공격자에게 세션 키를 결과 값으로 출력해 준다.
- Corrupt(Π_{ti}) : 이 질의를 통하여 공격자는 사용자 C_i의 내부 상태(state)값 혹은 비밀 값인 패스워드를 알 수 있다. 본 논문에서는 공격자가 이 질의를 통해 비밀 패스워드 값만 알 수 있다고 가정한다.
- Execute(Π_{ti}, Π_{si}) : 이 질의는 키 교환 프로토콜의 실행을 단순히 도청하는 소극적인 공격자를 모델화 한 것이다. 공격자는 이 질의를 통해 정직한 오라클 Π_{ti}과 Π_{si}의 프로토콜 실행 시 발생하는 대화 내용(transcript)을 결과 값으로 받을 수 있다.
- Test(Π_{ti}) : 이 질의는 공격자의 세션 키에 관한 지식을 측정하기 위해 사용되어진다. 랜덤 비트 값 b를 결정하기 위해 동전던지기가 행해진다. $b=1$이면 세션 키가 공격자에게 전달되어진다. $b=0$인 경우는 랜덤 값이 전달되어진다.

3.1.3 안전성 정의

서로 다른 패스워드를 사용한 그룹 환경의 패스워드 인증 및 키 교환 프로토콜에 대한 안전성을 정의하기 위해 이 프로토콜을 공격하는 공격자 A의 공격 이점(advantage)을 수학적으로 정의해야 한다. 다음의 실험을 고려해 보자. 공격자 A가 Test 질의를 했을 때, 실험은 동전던지기를 통해 랜덤 비트 값 b를 만든다. 만약 $b=1$이면 실험은 세션 키를 공격자에게 전달하고, $b=0$이면 실험은 랜덤 값을 공격자에게 전달한다. 이러한 실험에서는 공격자는 b값을 추측함으로써, 세션 키를 알려할 것이다. Succ를 공격자가 위의 b값을 정확히 추측한 사건이라 가정했을 때 프로토콜의 세션 키 안전성을 공격하는 공격자 A의 이점은 다항식 시간 T이내에 다음과 같이 정의된다.

$$Adv_P^{pake}(A,T) = 2\Pr[Succ] - 1$$

만약 $Adv_P^{pake}(A,T)$가 모든 확률적 다항식 시간 공격자에 대해서 무시할 수 있는 확률로 표현되어질 때 주어진 프로토콜 P는 안전하다라고 말한다.

〈표 2〉 표 기

표 기	의 미
Ui, Alice, Bob, Carole, David, Eric	정직한 사용자 혹은 클라이언트 (단, $1 \le i \le n$)
ID(Ui),ID(A), ID(B), ID(C), ID(D), ID(E)	사용자 Ui, Alice, Bob, Carole, David, Eric의 identity(단, $1 \le i \le n$)
Upi, pwa, pwb, pwc, pwd, pwe	Ui, Alice, Bob, Carole, David, Eric의 패스워드
Ex	패스워드 X를 이용한 대칭키 암호화
H_1, H_2	암호학적 해쉬 함수(예, SHA-1) H_1, H_2 : $\{0,1\}^* \rightarrow \{0,1\}^l$
$ticket_G$	사용자가 Clients에 속한 사용자들과 그룹 서비스를 받기 위한 티켓

4. 프로토콜 제안

본 장에서는 서로 다른 패스워드를 가진 그룹 환

경의 사용자들 간의 상호 인증 및 키 교환 프로토콜을 제안한다. 우선, 프로토콜 설명을 위해 <표 2>에 있는 표기를 사용한다. 프로토콜의 사용자수를 5명으로 가정하고 사용자와 서버의 집합을 각각 *Clients* = {ID(U1), ID(U2), ···, ID(Un)}, *Server* = {S1, S2, ···, Sn}로 정의하자. 본 논문에서는 오직 정적인 Clients와 Server집합을 가정하며 구성원의 입·출입을 허용하는 동적인 환경은 고려하지 않는다. 단, 각각의 서버는 안전한 방법을 통해 공통의 키 K를 미리 사전에 공유하고 있음을 가정한다.[5] PKI를 이용해서 그룹 사용자들 간에 안전하게 키를 공유하는 방법이 많이 제안되었으며 본 프로토콜에서는 그 방법들 중 하나를 선택해서 키를 공유한다.

4.1 그룹 환경 상호 인증 및 키 교환 프로토콜 설명

프로토콜의 전체적인 과정은 티켓 발급, 티켓 전달, 키 형성단계로 이루어진다. (1)~(3) 단계는 사용자 U1이 다른 사용자들 U2, ···, Un들과 인증 및 키 교환 서비스를 받기 원한다는 것을 알리고 그에 대한 서비스 티켓을 부여 받는 과정이다. (4) 단계는 해당 티켓을 사용자들에게 전달한다. (5)~(7) 단계에서 사용자들은 티켓을 이용해서 각각의 서버와 인증을 수행함과 동시에 공통의 키 k를 얻게 된다. 자세한 프로토콜 절차 및 설명은 아래에 설명하였다. 본 논문에 있는 수학 연산은 군(group)의 위수가 매우 큰 소수 q인 그룹 G_q군에서 이루어진다. 이 군은 먼저 큰 소수 p를 선택하고, p-1의 소인수 중 q를 선택하여 G_q군을 생성한다. 지수요소는 모두 법 q (Z_q^*)에서 이루어지고, 나머지 연산은 법 p에서 이루어진다. 랜덤 값 x, b, c, d, e, x_b, x_c, x_d, $x_e \in Z_q^*$에 대해서 표기의 간략화를 위해 중간 계산 값들에 대한 표기를 $E_{x_i} = E_{pwa}(g^{x_i})$ $E_{S_i} = E_{pwa}(g^{s_i})$, $E_{R_i} = E_{R_i}(k,\ ID(Ui)$ *Clients*)로 정의한다.

(1) U1은 랜덤 하게 $x_1 \in Z_p^*$을 선택한 다음, $E_{pwa}(g^{x_1})$을 계산하여 S1에게 *Clients* 집합에 속한 ID(S1), ···, ID(Sn)들을 함께 보낸다.

(2) S1은 $E_{pwa}(g^{x_1})$를 복호화 하여 g^{x_1}를 얻는다. 그리고 s_1, $k \in Z_p^*$을 랜덤 하게 선택해서 $E_{S_1} = E_{pwa}(g^{s_1})$ 값과 중간 키 값 $R_1 = H_1(Clients \| g^{x_1 s_1})$을 계산한다. 그 다음 $E_{R_1} = E_{R_1}(k, Clients)$을 R_1을 이용해서 계산한다. 또한 S1은 티켓의 유효기간 L을 정한 다음 $ticket_G = E_K(k, Clients, L)$를 계산한다. 그 후 E_{S_1}, E_{R_1}, $ticket_G$를 U1에게 전달한다.

(3) E_{y_1}, E_{R_1}, $ticket_G$를 받은 U1은 E_{s_1} 값을 복호화한 후에, R_1값을 계산한다. 그 후 R_1을 이용해서 $E_{R_1} = E_{R_1}(k,\ Clients)$을 복호화한 후에 k값 및 *Client*값을 복호화 해 낸다. *Clients* 값이 맞으면, U1은 S1을 인증하게 된다. U1은 중간 키 R_1에 대한 확인 값으로 $E_{R_1}(g^{x_1})$을 S1에게 전달하게 되는데, S1은 이 값을 복호화 해서 g^{x_1}값과 확인 후 맞으면, S1을 인증한다.

(4) U1는 통신을 원하는 *Clients* 그룹에게 S1로부터 받은 $ticket_G$를 전달한다.

(5)~(7) $ticket_G$를 받은 U2, ···, Un은 U1이 했던 (1)~(4) 과정을 동일하게 반복해서 공통의 k값을 얻게 된다. 공통의 k값을 이용해서 U2, ···, Un은도 동일한 방법으로 k를 이끌어낸 후 공통의 세션 키 $sk = H_2(Clients \| k)$를 계산한다.

4.1 5자간 그룹 환경 상호 인증 및 키 교환 프로토콜 설명

프로토콜에 참여하는 사용자들이 5명으로 가정

5) 키 K는 이후 티켓을 암호화 하는데 사용된다.

했을 때의 프로토콜을 예를 들어 설명하려 한다. 즉, *Clients* = {Alice, Bob, Carole, David, Eric}, *Server* = {ServerA, SeverB, ServerC, ServerD, ServerE}일 경우의 키 교환 및 상호 인증 프로토콜을 다음과 같이 구성할 수 있다([그림 4] 참조). 프로토콜에 사용되는 표기를 <표 3> 같이 요약하였다.

(1) U1은 랜덤 하게 $x \in Z_p^*$를 선택한 다음, $E_{pwa}(g^x)$를 계산하여 ServerA에게 *Clients* 집합에 속한 ID(A), ID(B), ID(C), ID(D), ID(E)들을 함께 보낸다.

〈표 3〉 암호화 표기 정의

$E_x = E_{pwa}(g^x)$,	$E_b = E_{pwb}(g^b)$,	$E_c = E_{pwb}(g^c)$,
$E_d = E_{pwb}(g^d)$,	$E_e = E_{pwb}(g^e)$,	$E_{x_b} = E_{pwb}(g^{x_b})$,
$E_{x_c} = E_{pwb}(g^{x_c})$,	$E_{x_d} = E_{pwb}(g^{x_d})$,	$E_{x_e} = E_{pwb}(g^{x_e})$,
$E_{R_b} = E_{R_b}(k,\ ID(B),\ Clients)$,		
$E_{R_c} = E_{R_c}(k,\ ID(C),\ Clients)$,		
$E_{R_d} = E_{R_d}(k,\ ID(D),\ Clients)$,		
$E_{R_e} = E_{R_e}(k,\ ID(E),\ Clients)$		

(2) ServerA는 $E_{pwa}(g^x)$를 복호화하여 g^x를 얻는다. 그리고 $y,\ k \in Z_p^*$을 랜덤 하게 선택해서 $E_y = E_{pwa}(g^y)$ 값과 중간 키 값 $R = H_1(Clients \| g^{xy})$을 계산한다. 그 다음 $E_R = E_R(k,\ Clients)$을 R을 이용해서 계산한다. 또한 ServerA는 티켓의 유효기간 L을 정한 다음 $ticket_G = E_K(k,\ Clients,\ L)$를 계산한다. 그 후 E_y, E_R, $ticket_G$를 Alice에게 전달한다.

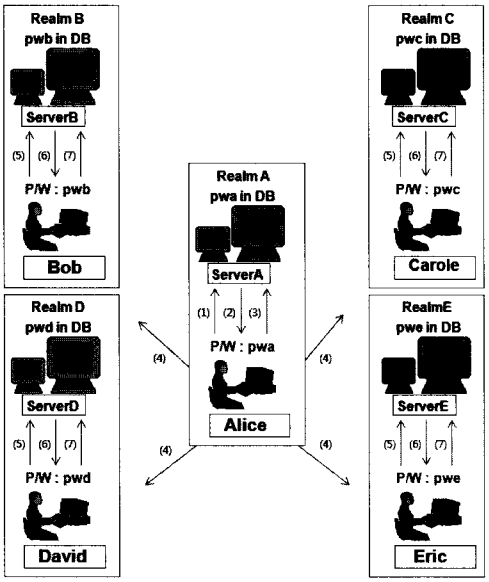
(3) E_y, E_R, $ticket_G$를 받은 Alice는 E_y값을 복호화 한 후에, R값을 계산한다. 그 후 R을 이용해서 $E_R = E_R(k,\ Clients)$을 복호화한 후에 k값 및 *Client*값을 복호화해 낸다. *Clients* 값이 맞으면, Alice는 ServerA를 인증하게 된다. Alice는 중간 키 R에 대한 확인 값으로 $E_R(g^x)$을 ServerA에게 전달하게 되는데, ServerA는 이 값을 복호화 해서 g^x값과 확인 후 맞으면, Alice를 인증한다.

(4) Alice는 통신을 원하는 *Clients* 그룹에게 Server A로 부터 받은 $ticket_G$를 전달한다.

(5)~(7) $ticket_G$를 받은 Bob, Carole, Daivd, Eric은 Alice가 했던 (1)~(4) 과정을 동일하게 반복해

티켓 발급 단계		
(1)	U1 ⇒ S1	E_{x_1}, Clients
(2)	S1 ⇒ U1	E_{S_1}, E_{R_1}, $ticket_G$
(3)	U1 ⇒ S1	$E_{R_1}(g^{x_1})$
티켓 전달 단계		
(4)	U1 ⇒ U2	$ticket_G$
(4)	U1 ⇒ U3	$ticket_G$
(4)	U1 ⇒ U4	$ticket_G$
(4)	U1 ⇒ U5	$ticket_G$
키 형성 단계 (U2)		
(5)	U2 ⇒ S2	E_{x_2}, Clients, $ticket_G$
(6)	S2 ⇒ U2	E_{S_2}, E_{R_2}
(7)	U2 ⇒ S2	$E_{R_2}(g^{x_2})$
키 형성 단계 (U3)		
(5)	U3 ⇒ S3	E_{x_3}, Clients, $ticket_G$
(6)	S3 ⇒ U3	E_{S_2}, E_{R_2}
(7)	U3 ⇒ S3	$E_{R_2}(g^{x_2})$
키 형성 단계		
	…	…
	…	…
	…	…
키 형성 단계 (Un)		
(5)	Un ⇒ Sn	E_{x_n}, Clients, $ticket_G$
(6)	Sn ⇒ Un	E_{S_n}, E_{x_n}
(7)	Un ⇒ Sn	$E_{R_n}(g^{x_n})$

[그림 5] 그룹 환경의 상호 인증 및 키 교환 프로토콜



[그림 4] 그룹 환경의 상호 인증 및 키 교환 서비스 흐름도 (n=5)

서 공통의 k값을 얻게 된다. Bob의 경우만 살펴보면, 우선 $b \in Z_p^*$를 랜덤 하게 뽑아서 $E_b = E_{pwb}(g^b)$를 계산하고 *Clients*, $ticket_G$과 함께 ServerB에게 전달한다. ServerB는 $x_b \in Z_p^*$를 뽑고 $E_{x_b} = E_{pwb}(g^{x_b})$를 계산한다. 그 후 E_b를 복호화 하여 Bob과 중간 키 값 $R_b = H_1(g^{bx_b})$값을 계산한다. 그 후 $ticket_G$를 복호화 하여 k값을 얻고 이를 R_b로 암호화 한 값 $E_{R_b} = E_{R_b}(k,\ D(B), Clients)$을 E_b 값과 함께 Bob에게 전달한다. Bob은 $E_{pwb}(g^{x_b})$를 복호화해서 중간 키 값 R_b을 얻게 되고 이를 이용해서 공통의 키 값 k를 복호화 해 낸다. 상호 인증을 위해 $E_{R_b}(g^{x_b})$를 계산해서 ServerB에게 전달하고 ServerB는 복호화 한 값이 자신이 보낸 값 g^{x_b}와 동일한지 검사한다. Carole, David, Eric도 동일한 방법으로 k를 이끌어낸 후 $sk = H_2(Clients \| k)$를 계산한다.

4.2 프로토콜 분석

통신 프로토콜의 효율성을 측정할 때에 중요한

티켓 발급 단계	
(1) Alice ⇒ ServerA	E_x, Clients
(2) ServerA ⇒ Alice	E_y, E_R, $ticket_G$
(3) Alice ⇒ ServerA	$E_R(g^x)$
티켓 전달 단계	
(4) Alice ⇒ Bob	$ticket_G$
(4) Alice ⇒ Carole	$ticket_G$
(4) Alice ⇒ Daivd	$ticket_G$
(4) Alice ⇒ Eric	$ticket_G$
키 형성 단계 (Bob)	
(5) Bob ⇒ ServerB	E_b, Clients, $ticket_G$
(6) ServerB ⇒ Bob	E_{R_b}, E_{x_b}
(7) Bob ⇒ ServerB	$E_{R_b}(g^{x_b})$
키 형성 단계 (Carole)	
(5) Carole ⇒ Serverc	E_c, Clients, $ticket_G$
(6) ServerC ⇒ Carole	E_{R_c}, E_{x_c}
(7) Carole ⇒ ServerC	$E_{R_c}(g^{x_c})$
키 형성 단계 (David)	
(5) David ⇒ ServerB	E_d, Clients, $ticket_G$
(6) ServerD ⇒ David	E_{R_d}, E_{x_d}
(7) David ⇒ ServerB	$E_{R_d}(g^{x_d})$
키 형성 단계 (Eric)	
(5) Eric ⇒ ServerB	E_e, Clients, $ticket_G$
(6) ServerE ⇒ Bob	E_{R_e}, E_{x_e}
(7) Eric ⇒ ServerB	$E_{R_e}(g^{x_e})$

[그림 6] 그룹 환경의 상호 인증 및 키 교환 프로토콜

요소는 계산량과 통신 라운드 수이다. 그 중 통신 라운드 수는 모든 통신비용을 결정하므로 가장 중요한 요소이다. 제안된 프로토콜의 계산량과 통신 라운드수는 참여자의 수에 선형적으로 의존한다. 참여자 수를 N이라 했을때 프로토콜 수행을 위해 필요한 통신 라운드는 3N+1이며 복잡도는 O(N)

으로 수렴된다. 지수승을 계산하는데 필요한 계산량 역시 참여자의 수에 의존하므로 O(N)으로 수렴된다. 제안된 프로토콜은 키를 만들기 위해 티켓 형태로 공통의 값을 참여자에게 보내고 그 값을 기반으로 하여 궁극적인 공통의 세션 키를 만들게 되므로 선형적인 통신, 계산 복잡도를 가지게 된다. 이를 분석하여 다음 표로 정리하였다.

〈표 4〉 제안하는 방식의 효율성 분석

	NR	NC	NE	NCR
횟수 분석	3N+1	4N	4N	N-1
점근적 분석	O(N)	O(N)	O(N)	O(N)

주) NR : 총 라운드 수, NC : 지수승, NE : 암호화 생성, NCR : 영역 간 라운드 수.

5. 안전성 분석

제 3장에서 정의한 대로 안전성을 분석한다. 정보보호 프로토콜의 효율적인 설계보다 더욱 중요한 것은 정보보호 프로토콜의 안전한 설계이다. 안전성의 정의는 해당 정보보호 프로토콜마다 다르다. 각 프로토콜의 목적에 맞는 공격자의 능력과 범위가 명확히 정의되고 이에 맞는 안전성 정의가 이루어진다. 그러므로 암호 프로토콜의 설계 후 반드시 검증해야 할 사항은 암호 프로토콜의 안전성이 증명이 되느냐의 여부이다. 안전성 증명은 암호학적 문제들에 대한 귀류법으로 이루어진다. 즉, 주어진 프로토콜의 안전성이 3장에서 정의된 공격자에 의해서 높은 확률로 깨어진다고 가정한다면, 공격자는 그 확률을 이용해서 어렵다고 가정한 문제가 쉽게 풀림을 보이려 할 것이다. 만약 높은 확률로 그 문제가 풀릴 수 있는 알고리즘을 구축할 수 있으면 이는 처음에 가정했던 계산적 가정 (문제가 풀리지 않음)에 위배되는 결과를 낳게 되므로, 역으로 프로토콜이 안전하다고 주장할 수 있다. 바꾸어 말하면, 정의된 공격자가 주어진 암호학적 어려운 문제(혹은 계산적 가정)를 다항식 시간(polynomial time) 안에 풀지 못한다면, 주어진 암호 프로토콜의 안전성을 깰 수 없다는 것과 동일하다.

다음 절에 계산적 Diffie-Hellman (CDH) 가정에 의해 제안된 프로토콜이 안전함을 랜덤오라클 모델에서 증명한다. 안전성을 정의하기에 앞서 CDH 문제 및 가정에 대한 정의가 필요하다.

5.1 계산적 가정

정의 1 : [CDH 문제] CDH 문제는 g, g^x, g^y가 주어졌을 때 g^{xy}를 다항식 시간에 구하는 문제이다. 단, x, $y \in Z_q^*$을 만족한다. 다항식 시간 T 이내에 g^{ab} 값을 구하는 공격자를 A로 정의하고 다음의 실험을 고려해보자.

〈표 5〉 실험 $\mathrm{Exp}_A(k)$의 정의

Experiment $\mathrm{Exp}_A(k)$

$x \leftarrow G_1;\ X \leftarrow g^x,\ y \leftarrow G_1;\ Y \leftarrow g^y$

$R \leftarrow A(CDH = (X, Y))$

IF $R = g^{xy}$ then output 1

CDH에 대한 공격자 A의 이점을 다음과 같이 정의한다.

$$Adv_A^{cdh}(T,\ k) = \Pr[\mathrm{Exp}_A(k) = 1]$$

정의 2 : [CDH 가정] : 만약 공격자 A에 대해서 CDH 문제를 푸는 이점이 무시할 수 있는 확률로 표현되어 질 때 CDH 가정이 Z_q^*상에서 만족된다고 정의한다.

정의 3 : [랜덤 오라클] : 제안된 프로토콜에서 정의된 해쉬 함수 H_1, H_2는 항상 랜덤 값을 출력한다고 가정하고 이를 랜덤 오라클이라 정의한다. 그 동작 방식은 다음과 같다.

〈표 6〉 랜덤 오라클 H_1 동작 원리

랜덤 오라클 H_1	
질의 m H(m)의 답변	If $m \in H_{list}$, then $r \leftarrow \{0, 1\}^l$ and $H_{list} \leftarrow H_{list} \| (m, r)$ Otherwise, r is taken from H_{list}

〈표 7〉 H_{list}의 구성 및 정의

H_{list}		
입력	출력	의미
m_1	r_1	$H_1(m_1) = r_1$ m_1은 H_1에 의해 질의되었음
m_2	r_2	$H_1(m_2) = r_2$ m_2은 H_2에 의해 질의되었음
...	...	...

5.1 안전성 증명

공격자 A가 주어진 프로토콜을 주목할 만한 이점으로 깬다고 가정한다면, 그 공격자를 이용하여 CDH 문제를 깰 수 있음을 다음 정리를 통해 보인다.

정리 1 : 제안된 프로토콜 P를 다항식 시간 T이내에 q_s, q_{h_1}, q_{h_2}의 질의를 통해 높은 확률 ε로 깨는 공격자 A를 가정했을 때, 이를 이용해서 CDH를 풀 수 있는 알고리즘 Δ를 다음과 같은 확률로 구성할 수 있다.

$$Adv_{\Delta}^{cdh}(k) \geq \frac{\varepsilon}{2q_{h_1}}$$

단, q_s, q_{h_1}, q_{h_2}는 Send, H_1, H_2의 질의 수이다.

<증명> 입력으로 g^a, g^b를 받았다고 했을 때 CDH 문제를 푸는 공격자를 Δ라고 가정하자. Δ는 프로토콜을 공격하는 공격자 A를 적절히 이용하여 주어진 g^a, g^b에 대한 g^{ab}값을 구하도록 설계되어야 한다.

우선, 공격자가 세션 키 $sk = H_2(Clients \| k)$를 구할 수 있는 경우는 오직 티켓에 포함되어 있는 공통의 키 값인 k를 아는 경우이다. 그러므로 공격자 A가 프로토콜의 세션 키를 높은 확률로 구할 수 있다는 것은 공격자 A가 k를 높은 확률로 획득했음을 의미한다. k를 구할 수 있는 방법은 암복호화 키인 $R = H_1(Clients \| g^{xy})$을 구해서 E_R값들을 복호화하는 경우이다.[6] 랜덤 오라클 모델에서는 공격자 A가 R을 계산하기 위해서는 반드시 랜덤 오라클 H_1에게 $Clients \| g^{xy}$를 높은 확률로 질의했음을 뜻한다. 이러한 사실들을 이용하여 CDH 문제를 푸는 알고리즘 Δ를 다음과 같이 구성한다.

△ 알고리즘

Δ는 먼저 입력으로 받은 g^a, g^b를 프로토콜에 적절히 이용(simulation) 해야 한다. 즉, 제 3장에 정의된 능동적 혹은 수동적 공격자 A의 모든 질의들에 대해서 답변을 정확히 해주도록 구성해야 한다. 이는 공격자 A가 현재 Δ를 통해서 질의에 대한 답변을 제공 받는 것인지 혹은 실제 프로토콜인지 구별하지 못하는 것을 의미한다. 먼저 Δ는 패스워드 생성 알고리즘 GPW를 통해서 모든 사용자들의 패스워드를 생성하고, R^U를 통해서 사용자와 패스워드를 등록한다.

- Send 질의 : Δ는 사용자의 모든 패스워드를 알고 있으므로 티켓 발급, 전달, 키 형성 단계의 모든 과정들을 직접 수행 할 수 있다. 단, Δ는 입력 받은 g^a, g^b를 적절히 Send 답변으로 이용해야 한다. 티켓 발급 단계에서의 답변과정을 살펴보자. [그림 5]의 (1)에서, Δ는 패

6) R_b, R_c, R_d, R_e를 통해서도 k를 구할 수 있지만 모두 랜덤오라클 H_1을 통해서만 만들어지므로 R의 경우만 분석한다.

스워드와 입력 받은 g^a를 이용하여, $E_x = E_{pwa}(g^a)$를 계산할 수 있고, 궁극적으로 $E_x, Clients$를 생성하여 답변한다. (2)에서도 패스워드와 입력값 g^b를 이용해서 $E_y = E_{pwa}(g^b)$를 계산한다. $E_R, ticket_G$를 계산하기 위해서는 R의 계산이 필요한데, Δ는 랜덤 값 r을 발생시켜 이 값을 R로 대체한다. 그 후 Δ는 랜덤 키 값 k를 선택하고 $E_R = (k \| Clients)$ 값 및 $ticket_G$ 값을 직접 계산해서 공격자 A에게 답변한다. 공격자 A는 CDH 문제를 풀지 못하는 한 $R = r$값이 Δ에 의해 발생된 값인지 구분할 수 없으며, 실제 프로토콜을 통해서 답변 받은 값으로 인식하게 된다. 나머지 과정은 Δ가 프로토콜의 순서에 의해 랜덤 값과 패스워드를 이용해서 답변을 한다.

- Reveal, Execute, Corrupt 질의 : Reveal 질의에 대해서는 기존에 만들어진 세션 키를 리스트에 보관 한 후에 공격자 A가 요청하면 리스트로부터 해당 세션 키 값을 반환한다. Execute는 프로토콜의 내용을 얻기 위한 것이므로 이미 만들어진 메시지 값들을 보관한 후에 공격자의 질의 시 해당 값을 반환한다. Corrupt 질의 시 Δ는 이미 생성한 패스워드 값을 이용해서 반환한다.
- Test 질의 : 공격자 A가 Test 질의를 했을 때, Δ는 랜덤 비트 값 b를 결정하기 위해 동전던지기를 수행한다. $b = 1$ (앞면)이면, 실제 세션 키를 공격자에게 반환하고, $b = 0$ (뒷면)이면, 랜덤 키를 선택해서 반환한다.

이제 Δ 알고리즘의 성공 확률을 분석해 보자. 공격자 A가 정확히 중간 키 값 R을 얻기 위해 $Clients \| g^{ab}$를 H_1에게 질의하는 사건을 AskH라고 정의하자. 공격자 A가 높은 확률 ε로 Test 질의에 선택된 키를 실제 키인지 랜덤 키인지 정확히 추측한다고 가정한다면, 이는 곧 높은 확률로 $sid' \| g^{ab}$를 H_1에게 질의했음을 의미한다. 왜냐하면, H_1에게 질의를 하지 않고는 $R = H_1(Clients \| g^{ab})$를 알 수 없고 궁극적으로 세션 키 $sk = H_2(Clients \| k)$도 알 수 없기 때문이다.

공격자 A가 Pr[AskH]의 확률로 주어진 알고리즘 Δ의 입력 값 g^a, g^b에 해당하는 g^{ab}를 질의했다면, 그것을 랜덤 오라클 H_{list} 리스트에서 정확히 g^{ab}를 선택할 확률만큼 CDH 문제를 풀 수 있다. 그 확률은 $\frac{\Pr[AskH]}{q_{h_1}}$ 이다. 단 q_{h_1}은 H_i에게 질의한 총 횟수이다. 랜덤 오라클 모델에서는 Ask 사건 없이 $b = b'$가 발생할 확률은 정확히 1/2이다. 그러므로 $2\Pr[b = b' | \neg Ask] - 1 = 0$를 얻는다. 이를 이용해 조건부 확률로 ε를 분석하면 다음의 식을 얻는다.

$$\begin{aligned}\varepsilon &= Adv_P^{pake}(A, T) = 2\Pr[Succ] - 1 \\ &= 2\Pr[b = b' | \neg Ask]\Pr[\neg Ask] \\ &\quad + 2\Pr[b = b' | Ask]\Pr[Ask] - 1 \\ &\le 2\Pr[b = b' | \neg Ask] - 1 + 2\Pr[Ask] \\ &\le 2\Pr[Ask]\end{aligned}$$

이는 정리 1의 결과를 만족시킨다.

5.2 전방향 안전성 분석

제 3장에서 제안된 안전성 모델은 전방향 안전성의 개념을 포함하지 않는다. 전방향 안전성을 다음과 같이 별도로 정의하고 제안된 프로토콜이 전방향 안전성을 만족함을 보이려 한다.

정의 4 : [전방향 안전성(pefrect forward secrecy)] : 롱텀(long-term) 비밀 값(패스워드)의 분실이 롱텀 비밀 값 분실 이전과 이후에 생성된 프로토콜 P의 세션 키 분실을 의미하지 않는다면, 프로토콜 P는 전방향 안전성을 만족한다라고 말한다.

본 논문에서는 동적인(dynamic) 그룹에서의 안전성을 논하지 않는다. 즉, *Clients* 멤버들의 그룹 내의 출입(join, leave)을 허용하지 않고 항상 고정적인 정적인(static) 구성원들을 가정한다. 그러므로 동적인 그룹에서 사용자들의 입출입을 고려한 전방향, 후방향 안전성 개념이 아니라, 정의 4에 정의한대로, 정적인 그룹에서 롱텀(long term) 비밀 값을 분실했을 경우 공격자가 세션 키를 구할 수 없어야 함을 의미한다. 공격자 A가 Corrupt의 질의를 통해 사용자의 패스워드를 획득했다고 가정하자. 하지만, 세션 키는 $sk = H_2(Clients \| k)$로 만들어지므로 k값을 반드시 알아야 한다. 이 값을 알기 위해서는 암복호화 키인 $R = H_1(Clients \| g^{xy})$을 구해서 E_R값들을 복호화하는 경우이다. 하지만, 이 값은 CDH 값인 g^{xy} 값으로 구성되기 때문에 구할 수 가 없다. 그러므로, CDH 문제를 풀지 못하는 한 공격자 A는 전 방향 안전성을 깰 수 가 없다. 단, 공격자 A가 k값 혹은 R값을 추측하여 세션 키를 구할 가능성은 있다. 그 확률은 $1/2^n$이므로 무시할 수 있다. 그러므로 제안된 프로토콜은 CDH 가정에 의해 전방향 안전성을 만족한다.

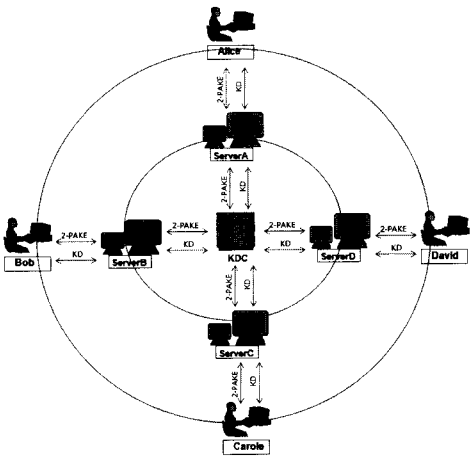
6. 포괄적인 구축 방법론에 대한 고찰

본 장에서는 이미 안전하게 구축되어 있는 양자간 패스워드기반 키 교환 (2-PAKE : 2-party password authenticated key exchange) 프로토콜 및 키 분배(KD : key distribution) 프로토콜을 이용한 포괄적인(generic) 구축 방법을 논의한다. 먼저, [그림 6]은 2-PAKE와 KD 프로토콜을 이용해서 서로 다른 영역에 존재하는 사용자들이 각자 고유의 패스워드를 이용해서 상호 인증 및 공통의 키를 발급 받는 총제적인 구축방법을 나타낸다.

KD 프로토콜은 두 개체 간에 사전에 공유된 비밀 키 값이 있음을 가정한다. 공유된 비밀 키 값을 가지고 대칭키 기술을 이용해 키 분배를 수행한다. 2-PAKE가 KD보다 먼저 수행되는 이유는 이러한 사전 공유 키를 2-PAKE를 통해서 형성하기 위해서이다.

6.1 구축 방법 설명

먼저, KDC(key distributin center)가 있음을 가정한다. KDC는 서버들을 관리하고 관련 키들을 분배하는 곳으로 서버들은 KDC에 항상 등록이 되어 있어야 한다.



[그림 6] 총제적인 구축 방법

(1) 우선, 각 영역에 있는 서버 ServerA, ServerB, ServerC, ServerD 들은 KDC와 각각 2-PAKE를 통해서 인증 및 공통의 키 sk_{AK}, sk_{BK}, sk_{CK}, sk_{DK}를 안전하게 형성한다.

(2) KDC는 형성된 키를 이용해서 키 분배 프로토콜을 실행시켜 모든 서버들에게 공통의 키 k를 분배하게 된다. 다시 말하면, KDC는 sk_{AK}, sk_{BK}, sk_{CK}, sk_{DK}를 이용해서 공통의 키 k를 모든 서버에게 각각 안전하게 전달한다.

(3) 각각의 서버는 자신의 사용자들과 2-PAKE를 통해서 또 다른 공통의 키 sk_{AS}, sk_{BS}, sk_{CS}, sk_{DS}를 형성한다. 서버는 KD를 통해서 전달 받은 키 k에 일방향 해쉬함수를 취한 값

$k' = H(k)$을 자신의 사용자들에게 전달한다. 각각의 사용자들은 키 k'를 이용해서 세션 키 $sk = H(Clients \| k')$를 계산한다.

6.2 안전성 분석

제안된 구축 방법의 안전성은 2-PAKE와 KD의 안전성에 의존한다. 암묵적으로 2-PAKE와 KD가 안전하다면, 전체적인 구축 방법이 안전하다라고 추측할 수 있다. 이에 대한 정밀한 증명을 위해서는 제안된 구축방법에 대한 정확한 안전성 모델과 2-PAKE와 KD의 안전성 수준을 고려해서 분석해야 한다. 이에 대한 안전성 모델 설립 및 수학적 증명은 향후 연구과제로 남긴다.

7. 결 론

서로 다른 패스워드를 이용한 인증 서비스 개념은 기존의 동일한 패스워드를 이용한 사용자-서버의 인증 모델의 한계점을 극복해주는 새로운 개념이다. 적어도 하나의 서버를 소유하고 있는 사용자들이라면 자신이 암기한 서로 다른 패스워드를 가지고 상호 인증 및 키 교환 서비스를 수행할 수 있다. 이는 사용자단에서 요구되는 사전 등록 및 인증에 필요한 절차를 간소화 시킬 수 있으며, 무엇보다 자신만 소유하고 있는 패스워드를 가지고 다른 사람과 인증 서비스를 수행한다는 측면에서 편리성을 제공한다. 그러므로 사용자 인증의 유연성을 강조하는 차세대 네트워크 환경에 필수적인 인증서비스로 각광 받을 것이다.

본 논문에서는 서로 다른 패스워드를 이용한 그룹 환경에서의 인증서비스 프로토콜을 제안하고 그 안전성을 증명하였다. 더 나아가 일반적이고, 총체적인 구축방법론 및 그 안전성에 대해서 고찰하였다. 제안된 프로토콜을 랜덤 오라클을 사용하지 않고 재설계될 수 있으면 이는 더욱 더 바람직한 결과가 될 것이다. 또한, 총체적인 구축론에 대한 안전성이 시급히 증명되어야 한다.

참 고 문 헌

[1] 김승희, 신경철, 제동국, 강숙양, 배정숙, 김재호, 박세권, 류승완, 유비쿼터스 정보화 사회에서 차세대 이동통신 융합서비스 제공을 위한 핵심 기술적 이슈 및 서비스 개발 프레임워크, 한국IT서비스학회지, 제7권, 제3호(2008), pp.215-237.

[2] Abdalla, M. and D. Pointcheval, "Interactive Diffie-Hellman Assumptions With Applications to Password-Based Authentication", In Proceedings of FC 2005, LNCS Vol.3570(2005), pp.341-356.

[3] M. Abdalla, P. Fouque, and D. Pointcheval, "Password-Based Authenticated Key Exchange in the Three-Party Setting", In Proceedings of PKC 2005, LNCS Vol.3386(2005), pp.65-84.

[4] Asokan, N. and P. Ginzboorg, "Key agreement in Ad-hoc networks", Computer Communications, Vol.23, No.17(2000), pp.1627-1637.

[5] M. Boyarsky, "Public-Key Cryptography and Password Protocols : The Multi-User Case", ACM Conference on Computer and Communications Security, 1999, pp.63-72.

[6] Byun, J. W. and D. H. Lee, "N-party Encrypted Diffie-Hellman Key Exchange Using Different Passwords", In Proc. of ACNS 2005', LNCS Vol.3531(2005), pp.75-90.

[7] Blake-Wilson, S. and A. Menezes, "Authenticated Diffie-Hellman Key Agreement Protocols", SAC 1998, LNCS 1556, 1999, pp.339-361.

[8] Bellare and Rogaway, "Provably secure session key distribution-the three party case", ACM symposium in theory of computing, 1995.

[9] Kim, H., D. Lee, and J. Lim, "Constant-Round Authenticated Group Key Exchange for Dynamic Groups", In Proceedings of Asiacrypt

2004, LNCS Vol.3329(2004), pp.245-259.

[10] Jeong, I., J. Katz, and D. Lee, "One-Round Protocols for Two-Party Authenticated Key Exchange", In Proceedings of ACNS 2004, LNCS Vol.3089(2004), pp.220-232.

[11] Bresson, E., O. Chevassut, D. Pointcheval, and J. J. Quisquater, "Provably authenticated group diffie-hellman key exchange", In proceedings of 8th ACM Conference on Computer and Communications Security, 2001, pp.255-264.

[12] Bresson, E., O. Chevassut, and D. Pointcheval, "Group diffie-hellman key exchange secure against dictionary attacks", In proceedings of Asiacrypt 2002, LNCS Vol.2501(2002), pp.497-514.

[13] Bresson, E., O. Chevassut, and D. Pointcheval, "Dynamic group diffie-hellman key exchange under standard assumptions", In proceedings of Eurocrypt 2002, LNCS Vol.2332(2002), pp.321-336.

[14] Bresson, E., O. Chevassut, D. Pointcheval, and J. J. Quisquater, "Provably authenticated group diffie-hellman key exchange in the dynamic case", In proceedings of Asiacrypt 2001, LNCS Vol.2248(2001), pp.290-309.

[15] Bellare, M., D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks", In proceedings of Eurocrypt 2000, LNCS Vol.1807(2000), pp.139-155.

[16] Bellare, M. and P. Rogaway, "Random oracles are practical : a paradigm for designing efficient protocols", In proceedings of the First ACM Conference on Computer and Communications Security, ACM, 1995.

[17] Bellare, M. and P. Rogaway, "Entity authentication and key distribution", In proceedings of Crypto 1993, LNCS Vol.773(1994), pp.232-249.

[18] Bellovin, S. and M. Merrit, "Encrypted key exchange : password based protocols secure against dictionary attacks", In proceedings of the Symposium on Security and Privacy, IEEE, 1992, pp.72-84.

[19] Boneh, D. and M. Franklin, "Identity-based encryption from the Weil pairing", Proc. of Crypto 2001, LNCS 2139, 2001, pp.213-229.

[20] Byun, J. W., I. R. Jeong, D. H. Lee, and C. Park, "Password-Authenticated Key Exchange between Clients with Different Passwords", In Proceedings of ICICS 2002, LNCS Vol. 2513(2002), pp.134-146.

[21] Byun, J. W. and D. H. Lee, "N-party Encrypted Diffie-Hellman Key Exchange Using Different Passwords", In proceedings of ACNS05', LNCS Vol.3531(2005), pp.75-90.

[22] Ding, Y. and P. Horster, "Undetectable on-line password guessing attacks", In ACM Operating Systems Review, Vol.29, No.4(1995), pp.77-86.

[23] Goldreich, O. and Y. Lindell, "Session-key generation using human passwords only", In proceedings of Crypto 2001, LNCS Vol.2139(2001), pp.408-432.

[24] Halevi, S. and H. Krawczyk, "Public-key cryptography and password protcols", In proceedings ACM Conference on Computer and Communications Security, ACM press, 1999, pp.63-72.

[25] Kim, J., S. Kim, J. Kwak, and D. Won, "Cryptoanalysis and improvements of password authenticated key exchange scheme between clients with different passwords", In Proceedings of ICCSA 2004, LNCS Vol.3044 (2004), pp.895-902.

[26] Lin, C., H. Sun, and T. Hwang, "Three-party encrypted key exchange : attacks and a solution", In ACM Operating Systems Review,

Vol.34, No.4(2000), pp.12-20.

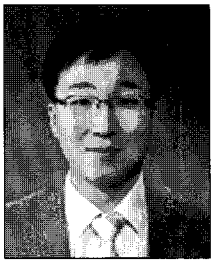
[27] Lin, Chun-Li., Hung-Min Sun, M. Steiner, and Tzonelih Hwang, "Three-party Encrypted Key Exchange Without Server Public-Keys", IEEE Communications Letters}, Vol.5, No.12 (2001), pp.497-499.

[28] Phan, R. C.-W., and B. Goi, "Cryptanalysis of an Improved Client-to-Client Password-authenticated Key Exchange (C2C-PAKE) Scheme", In proceedings of ACNS 2005, LNCS Vol.3531 (2005), p.33?9.

[29] Steiner, M., G. Tsudik, and M. Waider, "Refinement and extension of encrypted key exchange", In ACM Operation Sys. Review, Vol.29, No.3(1995), pp.22-30.

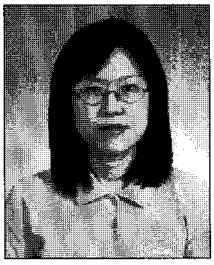
[30] Wang, S., J. Wang, and M. Xu, "Weakness of a password-authenticated key exchange protocol between clients with different passwords", In Proceedings of ACNS 2004, LNCS Vol.3089(2004), pp.414-425.

◈ 저 자 소 개 ◈



변 진 욱 (jwbyun@ptu.ac.kr)

고려대학교 전산학과에서 학사를 전공하고 고려대학교 정보보호대학원에서 정보보호 전공으로 석/박사 학위를 취득 후 런던대학교 ISG 연구소에서 박사후연구원으로 수학하였다. 2008년부터 평택대학교 정보통신학과에 전임강사로 재직 중이다. 개인정보보호, 프라이버시 보호 기술, 인증, 키 교환 알고리즘 등이 주 관심분야이다.



이 수 미 (smlee@fsa.or.kr)

고려대학교 정보보호대학원에서 정보보호 전공으로 석/박사를 취득하였다. 현재 금융보안연구원에서 인증 및 각종 금융보안과 관련한 연구를 수행 중이다. 주 관심분야는 키 교환 및 인증, 금융 정보보호 프로토콜 등이다.

이 동 훈 (donghlee@korea.ac.kr)

고려대학교에서 경영학을 전공하였고 Oklahoma 대학에서 전산학으로 석/박사를 취득하였다. 현재 고려대학교 정보보호경영전문대학원 교수로 재직 중이고 대학원 부원장을 역임하고 있다. 또한 고려대학교 BK21 유비쿼터스 정보보호사업단 단장을 역임하고 있다. 주 관심분야는 정보보호 프로토콜 및 임베디드 소프트웨어이다.