패스워드 기반의 효율적인 키 교환 프로토콜
(A Password-based Efficient Key Exchange Protocol)

이 성 운 † 김 현 성 †† 유 기 영 †††
(Sung-Woon Lee) (Hyun-Sung Kim) (Kee-Young Yoo)

요 약 본 논문에서는 작은 패스워드만을 이용하여 안전하지 않은 통신상에서 사용자와 서버간에 서로를 인증하고 세션키를 공유하기 위한 새로운 키 교환 프로토콜을 제안한다. 제안된 프로토콜의 안전성은 이산대수 문제와 Diffie-Hellman 문제의 어려움, 그리고 해쉬 함수의 암호학적 강도에 기반을 두고 있으며 패스워드 추측 공격, 중간 침입자 공격, Denning-Sacco 공격, 그리고 Stolen-verifier 공격에 안전하며, 완전한 전방향 보안성을 제공하도록 설계되었다. 더욱이, 구조가 간단하고 참여자들 사이에 병렬 처리가 가능하기 때문에 기존에 잘 알려진 프로토콜들과 비교하여 효율적이다.

키워드 : 암호, 키 교환, 패스워드, 인증

Abstract In this paper, we propose a new key exchange protocol which authenticates each other and shares a session key between a user and a server over an insecure channel using only a small password. The security of the protocol is based on the difficulty of solving the discrete logarithm problem and the Diffie-Hellman problem and the cryptographic strength of hash function. The protocol is secure against the man-in-the-middle attack, the password guessing attack, the Denning-Sacco attack, and the stolen-verifier attack, and provide the perfect forward secrecy. Furthermore, it is more efficient than other well-known protocols in terms of protocol execution time because it could be executed in parallel and has a simple structure.

Key words : cryptography, key agreement, key exchange, password, authentication

1. 서 론

인터넷과 같은 개방된 통신 환경에서 통신 상대간에 안전한 통신을 하기 위해서는 전송될 정보를 암호화하여야 하며, 통신 상대자간에 암호화를 위해 공통으로 사용할 키를 공유해야 한다. 이때 통신 상대자간에는 정보를 교환하고 있는 상대가 실제 의도한 상대인지를 확인하는 인증 과정이 반드시 필요하다. 따라서 사용자 인증과 암호화키의 공유는 안전한 정보 교환을 위해 해결해야 할 중요한 문제이며, 이를 위해서 보다 효율적인 프로토콜 개발이 절실히 요구된다.

사용자 인증 방식은 인증의 기반이 되는 요소가 무엇이냐에 따라 다음과 같이 세 가지로 분류된다. 첫째, 목소리 식별, 망막 검사 등과 같이 사용자의 물리적인 특징을 이용하는 인증 방법, 둘째, ID 카드나 스마트카드 등과 같이 사용자가 소유한 물건을 통한 인증 방법, 셋째, 패스워드와 같이 사용자가 알고 있는 지식을 통한 인증 방법이다. 첫 번째와 두 번째 방식은 강력한 보안을 위해 사용되기는 하지만 그에 따르는 부가적인 하드웨어 비용이 크다. 반면 세 번째 방식은 별도로 필요한 장비가 없기 때문에 큰 비용을 들이지 않고도 쉽게 사용될 수 있어 많이 이용되고 있으며 패스워드 프로토콜들이 이에 해당된다. 그러나 낮은 엔트로피를 가지는, 즉 사람이 기억할 수 있는 패스워드를 이용해야 하므로 패스워드 추측공격에 취약할 수 있다.

패스워드를 이용하여 서로를 인증하고 키를 교환하는 프로토콜은 크게 두 종류로 분류될 수 있다[1]. 첫째는 동일 패스워드(Balanced password) 기반의 프로토콜로서 두 참여자는 같은 한 개의 패스워드를 사용하여 서로를 인증한다. 이 방식은 Peer-to-Peer 형태의 통신에는 효율적으로 사용될 수 있지만 클라이언트 서버 환경에 사용된다면 패스워드 파일이 공격자에게 노출될 경우 프로토콜의 안전성이 크게 떨어질 수 있다. 둘째는

· 이 논문은 2003년도 두뇌한국21사업에 의하여 지원되었음
† 비 회 원 : 경북대학교 컴퓨터공학과
staroun@hanafos.com
†† 종신회원 : 경일대학교 컴퓨터공학과 교수
kim@kiu.ac.kr
††† 종신회원 : 경북대학교 컴퓨터공학과 교수
yook@knu.ac.kr
논문접수 : 2003년 6월 24일
심사완료 : 2004년 3월 4일

패스워드 검증자(Verifier) 기반의 프로토콜로서 동일 패스워드 기반의 프로토콜과는 달리 한 참여자는 클라이언트로, 다른 한 참여자는 서버의 역할을 수행하는 환경에서 사용될 수 있다. 클라이언트는 패스워드를 사용하고, 서버는 클라이언트의 패스워드를 가공한 결과 값을 패스워드 파일에 미리 저장해두고 프로토콜 수행 중에 해당 클라이언트에 대한 인증을 위한 검증 데이타로 사용한다. 서버에 저장되는 패스워드를 가공한 정보를 검증자라고 한다. 이 방식은 서버의 패스워드 파일에 패스워드에 대한 검증자만을 저장하기 때문에 패스워드 파일이 노출되더라도 공격자는 이 검증자를 이용하여 직접 클라이언트로 위장할 수 없어야 한다. 그러나 검증자를 획득한 공격자는 서버로 위장할 수 있고, 많은 비용이 드는 사전 공격(Dictionary attack)을 수행하면 패스워드를 알아낼 수도 있다. 그러므로 이 논문에서 앞으로 사용하는 'Stolen-verifier 공격에 안전하다'는 의미는 다른 프로토콜들에서와 동일하게 '검증자를 얻는 공격자가 클라이언트로 직접 위장할 수 없다'는 것을 뜻한다. 이러한 검증자 기반의 프로토콜들로는 PAK-X[2], AMP[3], B-SPEKE[4], SRP[5], SNAPI-X[6], AuthA[7] 프로토콜 등이 있다.

본 논문에서는 패스워드 검증자를 기반으로 하는 새로운 키 교환 프로토콜을 제안한다. 이 프로토콜의 안전성은 이산대수 문제(Discrete Logarithm Problem)와 Diffie-Hellman 문제의 어려움, 그리고 해쉬 함수의 암호학적 강도에 기반을 두고 있다. 제안한 프로토콜은 중간 침입자 공격, 패스워드 추측 공격, Denning-Sacco 공격, Stolen-verifier 공격에 안전하며, 완전한 전방향 보안성(Perfect forward secrecy)을 제공한다. 또한 제안한 프로토콜은 구조가 간단하고 참여자들 사이에 병렬 처리가 가능하기 때문에 기존에 잘 알려진 프로토콜들과 비교하여 효율적이다.

본 논문의 구성은 다음과 같다. 2장에서는 새로운 검증자 기반의 키 교환 프로토콜을 제안하고 3장에서는 제안된 프로토콜에 대한 안전성을 분석한다. 4장에서는 기존의 프로토콜들과 비교하여 제안된 프로토콜들의 효율성을 분석한다. 마지막으로 5장에서는 결론을 맺는다.

2. 제안한 프로토콜

본 장에서는 사용자는 패스워드를, 서버는 그 패스워드의 검증자를 사용하여 서로를 인증하고 세션키를 공유할 수 있는 패스워드 검증자 기반의 키 교환 프로토콜을 제안한다. 제안한 프로토콜은 Diffie-Hellman 방식[8]의 키 교환을 수행한다.

2.1 표기 및 초기설정

본 절에서는 제안된 프로토콜들에서 사용될 용어와 표기법, 그리고 가정들을 설정하고 프로토콜이 시작하기 전에 참여자들 간에 동의할 내용을 기술한다.

표 1 프로토콜을 위한 표기

기호	의 미
A, B	각각 클라이언트와 서버의 식별자
id	클라이언트의 ID
g	곱셈군(Multiplicative group) Z_n^*의 생성자(Generator)
n	큰 소수
π	클라이언트에 의해 선택된 패스워드
π'	공격자에 의해 추측된 패스워드
v	서버에 저장되는 패스워드 검증자(Verifier)
t^{-1}	Z_n^*상에서 t의 역수
a, b	A와 B에 의하여 각각 선택된 Z_n^*의 임의의 원소
$h()$	안전한 일방향 해쉬 함수 (One-way hash function)
Y_X	프로토콜 수행 과정 중에 참여자 X에 의해 계산된 값
K	세션키
$\oplus$	비트 XOR (Exclusive-OR) 연산
$x \stackrel{?}{=} y$	x와 y값이 같은지를 비교
k	보안 파라미터

프로토콜의 두 참여자 클라이언트(A)와 서버(B)는 합법적인 참여자들이다. A와 B는 안전하게 Z_n^* 상의 생성자인 g와 큰 소수인 n을 미리 공유하고 있다고 가정한다. $h()$는 $\{0,1\}^* \rightarrow \{0,1\}^k$인 안전한 일방향 해쉬 함수[9]이다. 이 해쉬 함수는 랜덤 오라클[10]로써 동작한다고 가정한다. 보안 파라미터 k는 해쉬 함수의 출력 값의 비트 크기이며 전사(Brute-force) 공격을 막을 수 있을 만큼 충분히 큰 크기를 가져야 한다. $\{0,1\}^*$는 임의의 길이를 갖는 유한한 이진 문자열이고 $\{0,1\}^k$는 k의 길이를 갖는 이진 문자열을 나타낸다. 프로토콜의 참여자인 A와 B는 합법적인 사용자들이다. A는 패스워드 π를 소유하고 있고 B는 A의 id와 패스워드의 검증자인 $v = g^{h(A,B,\pi)}$를 패스워드 파일에 저장하고 있다고 하자. 간단한 표기를 위해 'mod n' 연산 표기는 생략하기로 한다.

2.2 프로토콜의 수행

A와 B가 세션키를 생성하기 원할 때 제안된 키 교환 프로토콜은 다음과 같이 수행된다.

단계 1. A는 먼저 자신만이 알고 있는 패스워드를 이용하여 서버가 저장하고 있는 검증자와 동일한 값인 $v = g^{h(A,B,\pi)}$를 계산한다. 그리고 임의의 정수 a를 선택하고 $X_A = g^a \oplus v$를 계산하여 자신의 id와 함께 B에게 전송한다.

단계 2. B는 A로부터 id와 X_A를 받은 후에 id를 이용하여 패스워드 파일로부터 A의 검증자 v를 검색한다. 그리고 임의의 정수 b를 선택하고 $X_B = (v)^b \oplus v$를 계산

하여 A에게 전송한다. 그리고 B는 A의 응답을 기다리는 대기 시간을 이용하여 $K_B = (X_A \oplus v)^b = g^{ab}$, $V_A' = h(X_B, K_B)$, 그리고 $V_B = h(X_A, K_B)$를 계산한다.

단계 3. A는 B로부터 X_B를 받은 후에 $X_A \stackrel{?}{=} X_B$인지를 검사한다. 두 값이 같다면 프로토콜은 종료한다. 그러나 두 값이 같지 않다면 $K_A = (X_B \oplus v)^{a \cdot h(A,B,\pi)^{-1}} = g^{ab}$와 $V_A = h(X_B, K_A)$를 계산하고 V_A를 B에게 전송한다. 그리고 A는 B의 응답을 기다리는 대기 시간을 이용하여 $V_B' = h(X_A, K_A)$를 계산한다.

단계 4. B는 A로부터 V_A를 받은 후에 $V_A \stackrel{?}{=} V_A'$를 검사한다. 두 값이 같다면 B는 K_A가 정확하다고 확신한다. 그리고 V_B를 A에게 전송하고 세션키 $K = h(K_B) = h(g^{ab})$를 계산한다.

단계 5. A는 B로부터 V_B를 받은 후에 $V_B \stackrel{?}{=} V_B'$를 검사한다. 두 값이 같다면 A는 K_B가 정확하다고 확신한다. 그리고 A는 세션키 $K = h(K_A) = h(g^{ab})$를 계산한다.

제안한 프로토콜의 수행을 간단히 요약하면 그림 1과 같다.

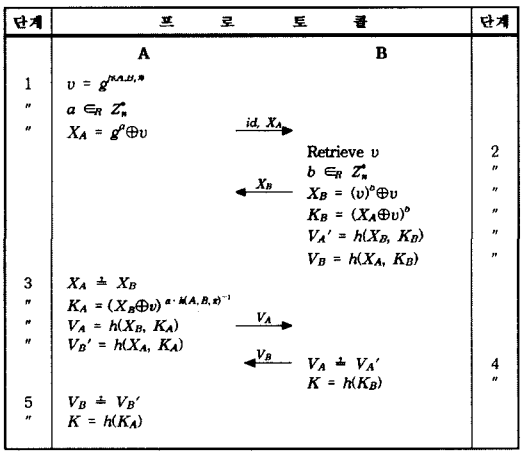

그림 1 제안한 프로토콜

프로토콜의 효율성을 높이기 위하여 프로토콜이 시작하기 전에 A는 $v = g^{h(A,B,\pi)}$와 $h(A,B,\pi)^{-1}$을 미리 계산할 수 있다. 이 값들은 프로토콜의 단계 1과 3에서 각각 사용된다. 그리고 단계 3에서 A가 $X_A \stackrel{?}{=} X_B$를 검사하는 이유는 공격자의 반송 공격을 막기 위함이다. 3.1절에서 이에 대해 더 자세히 알아보기로 한다.

제안된 프로토콜은 크게 두 과정, 즉 세션키 정보 생성 과정과 세션키 정보 검증 과정으로 나뉠 수 있다. 세션키 정보 생성 과정은 A와 B가 각각 자신의 정보와 상대방의 정보를 조합하여 Diffie-Hellman 값인 g^{ab}를 계산하는 과정이고 세션키 정보 검증 과정은 계산된 이 값이 정확한지를 검사함으로써 서로를 인증하는 과정이다. 제안된 프로토콜이 성공적으로 완료하면 A와 B는 같은 세션키인 $K = h(K_A) = h(K_B) = h(g^{ab})$를 공유하게 된다.

3. 안전성 분석

본 장에서는 먼저 안정성 분석을 위해 필요한 몇 가지 가정들과 정의들을 기술한다. 그리고 이들에 기반하여 제안된 프로토콜들의 안전성을 분석한다.

먼저, 2.1절에 기술된 바와 같이 전사 공격을 막기에 충분한 크기를 갖는 시스템 보안 파라미터 k를 가정하자. 그리고 임의의 사건에 대한 확률 Pr이 2^{-k}보다 작거나 같다면 그 확률은 무시할만하다고 하자. 사람이 기억할 수 있는 패스워드 π는 다항식 시간에 추측될 수 있는 낮은 엔트로피 $w(k)$ 값을 가진다. 이것은 공격자가 패스워드를 추측할 확률이 $1/2^{w(k)} >> 1/2^k$인 것을 의미한다. 그리고 프로토콜에 참여하는 참여자들 사이의 모든 통신은 논문 [11]에서와 같이 공격자의 제어 하에 있다고 가정한다. 즉 공격자는 통신 중간의 메시지들을 도청하거나 수정, 반송, 그리고 재전송할 수 있다. 심지어 정상적인 참여자로 위장해 프로토콜에 참여할 수도 있다. 공격자가 이러한 공격들을 통하여 부정확한 세션키 생성을 유도하거나 패스워드나 세션키를 알아낸다면 공격에 성공했다고 본다.

제안된 프로토콜들의 안전성은 다항식 시간에 풀기 어렵다고 알려져 있는 이산대수 문제와 Diffie-Hellman 문제[9]의 어려움에 근거한다. 두 가지 문제들은 다음과 같이 정의될 수 있다.

정의 1. 이산 대수 문제는 곱셈군 Z_n^*에서 생성자 g와 한 원소 g^a이 주어졌을 때 a를 계산하는 문제이다.

정의 2. Diffie-Hellman 문제는 곱셈군 Z_n^*에서 두 원소 g^a와 g^b이 주어졌을 때 g^{ab}를 계산하는 문제이다.

이 두 문제를 계산할 수 있는 확률이 각각 무시할만하다고 가정한다. 즉, $Pr <= 2^{-k}$이다. 제안된 프로토콜들은 2장에 기술된 바와 같이 공격자의 공격이 없다면 정확하게 동작한다는 것을 알 수 있다. 지금부터 앞에 기술된 가정과 정의들을 이용하여 제안된 프로토콜이 다양한 공격들에 대하여 안전함을 보이고자 한다.

3.1 중간 침입자 공격

키 교환 프로토콜은 공격자의 도청, 수정, 반송, 재전송, 위장 공격들에 대하여 세션키와 패스워드에 관한 정보를 노출시켜서는 안되며 잘못된 세션키의 공유를 탐지할 수 있어야 한다.

첫째로, 수동적인 공격을 고려해보자. 공격자는 전송

메시지들을 도청하여 $X_A = g^a \oplus g^{h(A,B,\pi)}$, $X_B = g^{b \cdot h(A,B,\pi)} \oplus g^{h(A,B,\pi)}$, $V_A = h(g^{b \cdot h(A,B,\pi)} \oplus g^{h(A,B,\pi)}, g^{ab})$, $V_B = h(g^a \oplus g^{h(A,B,\pi)}, g^{ab})$를 얻을 수 있다. 그러나 공격자가 이 값들을 획득한다 하더라도 π와, K를 계산할 수 있는 방법은 없다.

둘째로, 적극적인 공격자의 수정 공격을 고려하자. 공격자가 X_A와 X_B를 중간에서 수정하여 상대에게 전송한다면, 이 위조된 값들은 A와 B에 의해 K_A와 K_B를 생성하는데 각각 사용되게 된다. 그러나 A는 임의의 정수 a를 사용하여 K_A를 계산하고 B는 임의의 정수 b를 사용하여 K_B를 계산하기 때문에 K_A와 K_B의 값이 같게 될 확률은 무시할만하다. 결국, 이 공격은 검증 값들을 다르게 만들므로 검증단계에서 탐지될 수밖에 없다.

셋째로, 적극적인 공격자의 재전송 공격을 고려하자. 재전송 공격은 이전 세션의 전송 메시지들을 저장해 두었다가 이후 세션들에 이용하는 공격이다. 그러나 매 세션마다 각 참여자들은 새로운 임의의 난수 a와 b를 생성하여 사용한다. 공격자가 이 난수들을 알 수 있는 확률은 무시할 만하다.

넷째로, 적극적인 공격자의 반송 공격을 고려하자. 즉 공격자는 통신선로 중간에서 A가 B에게 보낸 X_A와 V_A를 A에게 되돌려 보내어 잘못된 세션키 생성을 유도하려 할 수 있다. 그러나 3단계에서 A는 B로부터 X_B를 받은 후에 $X_A \neq X_B$인지를 검사하기 때문에 이러한 공격은 성공할 수 없다.

다섯째로, 공격자는 합법적인 참여자로 위장하여 정상적인 방법으로 다른 합법적인 참여자와 세션키를 공유하려고 할 수 있다. 그러나 이러한 위장 공격은 공격자가 패스워드를 알지 못하기 때문에 검증 단계에서 탐지될 수밖에 없다.

결국 제안한 프로토콜들은 이와 같은 중간 침입자 공격들에 안전하다.

3.2 패스워드 추측 공격

패스워드 추측 공격은 온라인 패스워드 추측 공격과 오프라인 패스워드 추측 공격으로 나뉠 수 있다. 온라인 패스워드 추측 공격은 패스워드 인증 실패 횟수를 누적함으로써 쉽게 탐지되고 시도 횟수를 제한함으로서 쉽게 조치될 수 있다. 그러나 공격자는 안전하지 않은 통신상에서 메시지를 도청하거나 정당한 사용자로 가장하여 다른 정상 사용자와의 메시지 교환을 통해 발생하는 정보들을 저장해 두고 오프라인으로 패스워드에 관한 정보를 알아내려고 할 수 있다. 이러한 공격을 오프라인 패스워드 추측 공격이라 한다.

먼저 도청한 메시지만을 이용하는 수동적인 패스워드 추측 공격을 고려하자. 공격자는 메시지 X_A, X_B, V_A, V_B를 가로채 저장하고, 패스워드로 사용될 수 있는 π'를 추측한다. 그리고 π'을 도청한 값들에 적용하여 비교함으로 검증한다. 이를 모든 패스워드 범위에 대하여 반복 수행함으로써 추측한 π'가 참여자들이 사용하고 있는 정확한 π인지를 확인해야 한다. 그러나 제안된 프로토콜들에서는 전송 메시지인 X_A, X_B, V_A, V_B에 π'를 적용하여도 π'가 정확한지를 검증할 방법이 없다.

또한 공격자가 정당한 참여자로 위장한 적극적인 패스워드 추측 공격을 고려해 보자. 공격자가 A로 위장한다면 자신이 만든 a, g^a, $g^a \oplus g^{h(A,B,\pi)}$와 B로부터 받은 $g^{b \cdot h(A,B,\pi)} \oplus g^{h(A,B,\pi)}$를 얻을 수 있다. 그러나 이들을 이용해서는 π'가 정확한지를 검증할 방법이 없다. 그리고, 공격자가 B로 위장한다면 자신이 생성한 b, g^b, $g^{h(A,B,\pi)}$, $g^{b \cdot h(A,B,\pi)}$, $g^{b \cdot h(A,B,\pi)} \oplus g^{h(A,B,\pi')}$와 A로부터 받은 $g^a \oplus g^{h(A,B,\pi)}$, $h(g^{b \cdot h(A,B,\pi)} \oplus g^{h(A,B,\pi')}, (g^{b \cdot h(A,B,\pi')} \oplus g^{h(A,B,\pi')})^{a \cdot h(A,B,\pi)^{-1}})$ 값들을 얻을 수 있다. 그러나 이 값들을 이용해서도 π'가 정확한지를 검증할 방법이 없다. 그러므로 제안된 프로토콜들은 패스워드 추측 공격에 안전하다.

3.3 Denning-Sacco 공격

Denning-Sacco 공격은 세션키가 노출되었을 때 공격자가 패스워드에 관한 정보를 얻고자 하는 공격이다. 제안된 프로토콜들에서 공격자가 임의의 세션에서 도청을 통해 X_A, X_B, V_A, V_B를 얻었고, 세션키 $h(g^{ab})$가 공격자에게 노출되었다고 가정하자. 그러나 이 정보들로부터 패스워드를 구할 수 있는 방법은 없다.

3.4 Stolen-verifier 공격

Stolen-verifier 공격은 서버로부터 패스워드 검증자를 훔친 공격자가 직접적으로 합법적인 사용자를 가장하려는 공격을 의미한다. 제안된 프로토콜에서 서버에 저장된 패스워드 검증자는 $v = g^{h(A,B,\pi)}$이다. 이 자료를 훔친 공격자는 새로운 세션의 3단계에서 사용해야 하는 π를 직접적으로 계산할 수 없다. 그러므로 패스워드 검증자를 도난 당했을 때 위험이 감소된다고 볼 수 있다.

3.5 완전한 전방향 보안성

완전한 전방향 보안성을 제공하기 위해서는 패스워드가 공격자에게 노출되었다 할지라도 이전의 세션키들은 안전해야 한다. 제안된 프로토콜들에서 공격자에게 패스워드 π가 노출되었다고 하자. 공격자는 도청을 통해 X_A, X_B, V_A, V_B를 얻을 수 있다. 그러나 이 정보들로부터 세션키인 $h(g^{ab})$를 구할 수 있는 확률은 이산대수 문제와 Diffie-Hellman 문제의 어려움 때문에 무시할만하다.

4. 효율성 분석

이 장에서는 제안된 프로토콜의 효율성을 측정하기 위하여 지금까지 안전하다고 알려진 IEEE P1363.2 [1]에 제출된 패스워드 검증자 기반의 프로토콜들과 비교

한다.

키 교환 프로토콜들의 성능은 통신 부하와 계산 부하 면에서 측정될 수 있다. 통신 횟수는 통신 부하를 결정하는 기준이고 지수 연산 횟수, 해쉬 연산 횟수, 그리고 대칭키 연산 횟수는 계산 부하를 결정하는 기준들이다. 표 2는 이러한 성능 평가 기준들의 측면에서 제안된 프로토콜들의 성능을 기존에 잘 알려진 프로토콜들인 PAK-X[2], AMP[3], B-SPEKE[4], SRP[5], SNAPI-X[6], AuthA[7]들과 비교한 결과를 보여준다.

표 2에서 보는 바와 같이 제안된 프로토콜은 각 참여자가 수행해야 할 지수 연산 횟수, 해쉬 연산 횟수, 그리고 대칭키 연산 횟수에서 가장 효율적이다. 또한 프로토콜의 전체 수행시간에 가장 많은 영향을 미치는 지수 연산에 대한 병렬 수행 횟수를 고려해 보자. 'E(A:B)'는 두 참여들 사이에서 지수 연산의 병렬 수행을 의미한다고 하자. 즉, 이 표현은 한 참여자는 다른 참여자의 응답을 기다리는 동안 자신의 연산들을 수행할 수 있음을 의미한다. 제안된 프로토콜은 가장 작은 횟수를 갖는 AMP와 동일한 3E의 수행시간을 필요로 한다. 즉, $E(g^a : -)$, $E(- : (v)^b)$, $E((X_B \oplus v)^{a \cdot h(A,B,\pi)^{-1}} : (X_A \oplus v)^b)$. 여기서 '-'는 해당 참여자의 지수 연산 계산이 없음을 의미한다.

통신 횟수 측면에서 몇몇 프로토콜들(PAK-X, AuthA)은 통신비용을 줄이기 위해 3번의 메시지 교환을 통하여 프로토콜을 수행한다. 이 방법은 통신비용은 줄일 수 있지만 두 참여자 사이의 연산 수행을 직렬화시킨다. 즉 한 참여자는 다른 참여자로부터 응답을 받은 후에야 그 응답 메시지를 이용하여 자신의 다음 연산을 수행할 수 있다. 이로 인해 프로토콜의 전체 수행 시간은 길어질 수 밖에 없다. 그래서 본 논문에서는 제안된 프로토콜의 통신 횟수를 AMP 프로토콜 등과 같이 각 참여자들간에 병렬 수행이 가능하도록 하기 위하여 4번의 메시지 교환을 채택하였다. 즉 한 참여자는 가능한 빠른 시간에 다른 참여자가 필요로 하는 값만을 계산하여 전송하여 준 후 자신의 다른 연산들을 수행할 수 있다. 다른 연산들에 비해 비교적 많은 계산 시간을 요구하는 지수 연산을 사용해야 하는 키 교환 프로토콜에서 이러한 병렬 수행은 전체 성능에 중요한 영향을 미친다고 할 수 있다. 그러나 적은 통신 부하를 요구하는 응용들에서는 이러한 통신 횟수가 적은 프로토콜이 효과적일 수 있다. 이와 같은 환경에서 사용할 수 있도록 제안된 프로토콜을 3회의 통신을 수행하는 프로토콜로 변형하기는 매우 쉽다. 즉, 제안된 프로토콜의 연산들에 대해서는 아무런 변경 없이 각 참여자들이 $A \xrightarrow{id, X_A} B \xrightarrow{X_B, V_B} A \xrightarrow{V_A} B$ 형태로 통신을 하도록 통신 흐름만 변형하면 된다. 이 변형 프로토콜은 제안된 프로토콜에 비해 전체 수행 시간은 길어지지만 각 각 참여자가 수행해야 하는 각 연산의 수는 제안된 프로토콜과 동일하고 프로토콜의 안전성에 문제를 발생시키지 않는다.

표 2 기존 프로토콜들과의 효율성 비교

프로토콜 / 분석요인	통신 횟수	지수 연산 횟수			해쉬 연산 횟수			대칭키 연산 횟수
		A	B	병렬	A	B	병렬	
B-SPEKE	4	3	4	6	2	2	2	4
SRP	4	3	3	4	4	3	4	0
SNAPI-X	5	5	4	7	4	3	6	0
PAK-X	3	4	4	8	5	5	7	0
AMP	4	2	3	3	5	4	5	0
AuthA	3	3	3	6	4	4	7	4
Ours	4	2	2	3	3	3	3	0

5. 결 론

본 논문에서는 클라이언트는 패스워드를, 서버는 패스워드 검증자를 이용하여 서로를 인증하고 세션키를 공유하는 패스워드 검증자 기반의 새로운 키 교환 프로토콜을 제안하였다. 이 프로토콜은 중간 침입자 공격, 패스워드 추측 공격, Denning-Sacco 공격, 그리고 Stolen-verifier 공격에 안전하고, 또한 완전한 전방향 보안성을 제공하도록 설계되었다. 더욱이 제안한 프로토콜은 구조적으로 간단하고 병렬성을 제공하기 때문에 기존의 잘 알려진 프로토콜들과 비교하여 좋은 효율성을 가진다.

참 고 문 헌

[1] IEEE. Standard Specifications for Public Key Cryptography, IEEE1363, 2002.

[2] V. Boyko, P. MacKenzie and S. Patel. "Provably Secure Password-Authenticated Key Exchange Using Diffie-Hellman," Advances in Cryptology-EUROCRYPT'2000, pp. 156-171, 2000.

[3] T. Kwon. "Ultimate Solution to Authentication via Memorable Password," Presented to IEEE P1363a, May 2000.

[4] D. Jablon. "Extended password key exchange protocols," WETICE Workshop on Enterprise Security, 1997.

[5] T. Wu. "Secure remote password protocol," Internet Society Symposium on Network and Distributed System Security, 1998.

[6] P. MacKenzie, S. Patel, and R. Swaminathan. "Password-authenticated key exchange based on RSA." In ASIACRYPT2000.

[7] M. Bellare and P. Rogaway, "The AuthA protocol for password-based authenticated key exchange," Presented to IEEE P1363a, March 2000.

[8] W. Diffie, M. E. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, Vol.IT-22, No.6, pp.644-654, 1976.
[9] D. R. Stinson, Cryptography Theory and Practice, CRC, 1995.
[10] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," In ACM security 93, pp.62-73, 1993.
[11] M. Bellare and P. Rogaway, "Entity Authentication and Key Distribution," Advances in Cryptology-CRYPTO'93, Vol. 773, pp.232-249, 1994.

이 성 운
1993년 8월 전남대학교 전산통계학과 학사졸업. 1996년 8월 전남대학교 전산통계학과 석사졸업. 2001년 3월~현재 경북대학교 컴퓨터공학과 박사과정. 관심분야는 정보보호, 암호학, 네트워크 보안

김 현 성
1996년 경일대학교 공과대학 컴퓨터공학과(공학사). 1998년 경북대학교 공과대학 대학원 컴퓨터공학과(공학석사). 2002년 경북대학교 공과대학 대학원 컴퓨터공학과(공학박사). 2002년~현재 경일대학교 공과대학 컴퓨터공학과 교수. 관심분야는 암호연산, 병렬처리, 암호화 프로토콜

유 기 영
정보과학회논문지 : 정보통신
제 31 권 제 3 호 참조