논문 18-43-02-11 The Journal of Korean Institute of Communications and Information Sciences '18-02 Vol.43 No.02
https://doi.org/10.7840/kics.2018.43.2.281

중간자공격에 안전한 무인증서 기반 인증 키 합의 프로토콜

박 진 현◆, 김 순 자°

Secure Certificateless Authenticated Key Agreement Protocol Against MITM Attack

Jin-Hyun Park◆, Soon-Ja Kim°

요 약

본 논문에서는 기존 타원 곡선 기반 무인증서 인증 키 합의 프로토콜이 중간자공격에 취약하다는 것을 증명하고 중간자공격에 안전한 무인증서 인증 키 합의 프로토콜을 제안한다. 제안 프로토콜은 타원 곡선 이산 대수 문제가 유효하다는 전제하에 중간자공격에 안전함을 증명하였다.

Key Words : Key Agreement, Authentication, ECDLP, MITM, Security

ABSTRACT

We show that the existing certificateless authenticated key agreement protocol based on the elliptic curve cryptography is vulnerable to the man-in-the-middle attack. This paper proposes a certificateless authenticated key agreement protocol resistant to this attack. We proved that the proposed protocol is resistant to the man-in-the-middle attack if the elliptic curve discrete logarithm problem holds.

Ⅰ. 서 론

중간자공격은 두 당사자가 송수신하는 메시지를 가로채고 변조 또는 위조함으로써 정상적인 통신을 방해한다. 중간자공격이 발생할 경우 두 당사자는 의도하지 않는 상대방과 통신을 하게 되며 이 사실을 두 당사자는 알 수 없다. 이 공격을 방지하기 위한 방법으로 서명 스킴을 이용한 방법이 있다. 만일 공개키 기반 키 합의 프로토콜일 경우 서명 스킴과 함께 인증서를 이용한다. 이 인증서는 서명을 검증하기 위해서 사용하는 키의 진위성을 판단하기 위해서 사용한다.

기존 타원 곡선 기반 무인증서 인증 키 합의 프로토콜의 경우 프로토콜 자체에 서명 알고리즘이 포함되어 있다[1,2]. 이 알고리즘은 사용자의 부분 공개키를 키 생성 센터(KGC)의 공개키를 이용하여 진위성을 검증한다. 기존 프로토콜들은 랜덤 오라클 모델에서 총 9가지 위협 상황을 대상으로 안전성을 검증하였다[1,2]. 이는 공격자가 전체 6개 비밀키 중 2개를 모르는 상황을 조합한 것이다. 이 때 각 사용자마다 최소 한 개의 비밀키는 비밀로 유지된다. 결과는 프로토콜의 안전성이 랜덤성에 의존하며 공식 또는 알고리즘을 사용함으로써 발생하는 문제가 없다는 것이다.

본 논문에서는 이들 프로토콜이 9가지의 위협 중 한 가지 위협 상황에서 중간자공격이 가능하다는 것을 보인다. 이는 공개 채널을 통해 전송되는 부분 공개키를 교체함으로써 가능한 공격이다. 이와 같은 공격이 가능하다는 것은 프로토콜에 포함된 서명 알고리즘이 정상적인 기능을 못한다는 것이다. 이에 중간자공격에 안전한 무인증서 인증 키 합의 프로토콜을 제안한다. 제안 프로토콜은 타원 곡선 이산 대수 문제(ECDLP)가 유효하다는 전제하에 중간자공격에 저항성을 가짐을 보인다.

Ⅱ. 관련 연구

2016년 Farouk 등은 He 등이 제안한 프로토콜을 기반으로 연산 효율을 개선한 무인증서 인증 키 합의 프로토콜을 제안하였다[1,2]. 본 장에서는 He등이 제안한 프로토콜을 살펴본 후 중간자 공격에 취약하다는 것을 보인다.

2.1 기존 프로토콜 리뷰

프로토콜은 3개의 노드가 참여하며 6개의 알고리즘으로 구성되어 있다. 세션키 생성을 원하는 두 개의 노드, 공개 및 부분 파라미터를 생성하는 KGC가 있으며 각 알고리즘은 특정 노드에서 실행된다. 각 알고

※ 이 논문은 2015학년도 경북대학교 복현학술연구비에 의하여 연구되었음.
◆ First Author : (ORCID:0000-0002-8012-0314)College of IT Engineering, Kyungpook National University, helloworld@knu.ac.kr, 정회원
° Corresponding Author : College of IT Engineering, Kyungpook National University, snjkimd@ee.knu.ac.kr, 종신회원
논문번호 : KICS2018-01-002, Received January 2, 2018; Revised January 11, 2018; Accepted January 12, 2018

리즘에 대한 설명은 다음과 같다[1].

1) Setup: KGC는 소수 p를 선택하고 유한체 상의 타원 곡선 E/F_p, 타원 곡선상의 점으로 구성된 덧셈 순환군 G와 생성원 P를 선택한다. 또한 서명과 세션 키 생성을 위해 두 개의 암호학적 해시 함수 H_1, H_2를 선택한다. 끝으로 자신의 비밀 키 $x \in Z_n^*$ 및 공개키 $MP = xP(= P + P + ... + P : x$번 수행)를 설정한다. 비밀값을 제외한 파라미터는 모두 공개한다.

2) Partial-Private-Key-Extract: 노드 등록 요청 시 KGC는 난수 r을 선택 후 $PP = rP$, $s = H_1(ID, PP)$ 그리고 $v = r + s \bullet x \bmod n$를 계산한다. PP와 v는 안전한 채널로 해당 노드에 전송한다.

3) Set-Secret-Value: 각 노드가 자체적으로 비밀키 $x \in Z_n^*$를 생성하는 단계이다.

4) Set-Private-Key: 각 노드는 x와 v를 자신의 비밀키로 설정한다.

5) Set-Public-Key: 각 노드는 자신의 비밀키 x를 이용하여 공개키 $SP = xP$를 계산한다.

6) Key-Agreement: 노드 N_i는 난수 t를 선택하고 임시 공개키 $TP = tP$를 계산한다. 이 후 (ID_i, PP_i, TP_i)를 공개 채널로 노드 N_j에 전송한다. 마찬가지로 N_j도 (ID_j, PP_j, TP_j)를 N_i로 전송한다. 최종 세션키는 $H_2(ID_i, ID_j, TP_i, TP_j, K_1, K_2, K_3)$호출을 통해서 생성한다. 여기서 K_i는 다음과 같다.

$$K_1 = (t_i + v_i)(TP_j + PP_j + H_1(ID_j, PP_j)MP) \quad (1)$$

$$K_2 = (t_i + x_i)(TP_j + SP_j) \quad (2)$$

$$K_3 = t_i TP_j \quad (3)$$

여기서는 N_i에 의한 식만 포함하였다. N_j의 경우 인덱스 i, j를 바꿈으로써 동일한 결과를 얻는다.

2.2 취약점 분석

기존 논문에서는 공격자가 3개의 비밀키 중 2개를 알고 있을 때 나머지 한 개의 값을 계산하는 문제가 ECDLP에 종속됨을 증명하였고 그 결과 프로토콜이 안전하다는 것을 주장하였으나 본 논문은 ECDLP를 우회한 중간자 공격이 가능함을 보인다. 우회가 가능한 이유는 부분 공개키 교체가 가능하기 때문이다. 취약점 분석을 위해 기존의 위협 상황과 동일한 가정을 한다. 이는 공개키는 공격자에 의해 교체되었고 비밀키와 임시키도 노출된 상황이다.

공격자는 사전에 난수 $r_a \in Z_n^*$을 선택한 후 $PP_a = r_a P$, $s_a = H_1(ID_i, PP_a)$를 계산한다. 이는 부분 공개키를 변조한 상황에 해당된다. 노드 N_i가 노드 N_j로 (ID_i, PP_i, TP_i)를 전송할 때 공격자는 이 메시지를 가로채고 동시에 $TP_a = -s_a MP$를 계산한다. 그리고 (ID_i, PP_a, TP_a)를 N_j에 전송한다. N_j가 N_i로 보내는 메시지도 동일하게 값을 위조한다. 여기서는 N_j가 계산한 세션키와 공격자가 계산한 세션키가 같음을 보인다. 현재 위협 상황은 공격자가 비밀키 x와 임시키 t를 알고 있는 경우이므로 K_1만 일치한다면 동일한 세션키를 계산할 수 있다. 따라서 여기서는 K_1값의 일치 여부만 확인한다. N_j가 계산한 K_1은 식 (4)와 같다.

$$K_1 = (t_j + v_j)(TP_a + PP_a + H_1(ID_i, PP_a)MP) \quad (4)$$

여기서 SP_a는 공격자에 의해 교체된 N_j의 공개키이다. 공격자가 계산한 K_1'은 식 (5)와 같다.

$$K_1' = r_a TP_j + r_a(PP_j + H_1(ID_j, PP_j)MP) \quad (5)$$

식 (4)는 $K_1 = (t_j + v_j)(-s_a MP + PP_a + H_1(ID_i, PP_a)MP) = (t_j + v_j)PP_a = r_a TP_j + r_a(PP_j + H_1(ID_j, PP_j)MP)$와 같이 변형 되며 이는 식 (5)의 K_1'과 동일하다. 공격의 결과 각 노드는 공격자와 동일한 세션키를 생성한다. 이 공격 방식을 이용하면 3개의 식 중 최소 하나는 변조가 가능하다. Amr등이 제안한 프로토콜도 동일한 취약점이 발견되는데 이는 이들이 제안한 프로토콜은 세션키 생성을 위한 식이 하나밖에 없기 때문이다.

Ⅲ. 제안하는 프로토콜

이 장에서 중간자공격에 안전한 무인증서 기반 인증 키 합의 프로토콜을 제안한다. 다음은 각 단계에 대한 설명이다.

제안하는 프로토콜:

1) Setup: KGC는 기존 알고리즘과 동일한 방식으로 공개 파라미터를 생성하고 자신의 비밀키 ms와 공개키 msP를 생성한다. 또한 암호학적 해시 함수 H_0, H_1, H_2를 선택한다.

2) 비밀키, 공개키 생성: 각 노드는 자체적으로 비밀키 $ns \in Z_n^*$와 자신의 공개키 nsP를 생성한다.

3) 부분 공개키, 비밀키 생성: 각 노드의 등록 요청 시 KGC는 난수 r을 선택하고 부분 공개키 rP, 부분 비밀키 $nps = r + H_1(ID_i, H_0(rP, nsP))\, ns \bmod n$를 계산한다. rP, nps는 안전한 채널을 통해 해당 노드에 전송한다.

4) 세션키 생성: 각 노드는 (ID, rP, tP, nsP)를 상대방에 전송한다. 여기서 t는 임시 난수이다. 세션키 생성을 위해서 노드 i는 $npsP_j = rP_j + H_1(ID_j, H_0(rP_j, nsP_j))\, msP$를 계산한다. 그리고 식 (6), (7), (8)을 계산한다.

$$K_1 = (t_i + nps_i)(tP_j + npsP_j) \tag{6}$$

$$K_2 = (t_i + ns_i)(tP_j + nsP_j) \tag{7}$$

$$K_3 = (ns_i + nps_i)(nsP_j + npsP_j) \tag{8}$$

인덱스 i, j를 교체할 경우 노드 j가 계산하는 식이 된다. $abP = baP$가 성립하므로 각 노드는 동일한 K_1, K_2, K_3을 얻게 된다. 각 노드는 최종 세션키 $K = H_2(ID_i, ID_j, tP_i, tP_j, K_1, K_2, K_3)$를 생성한다.

정리: ECDLP가 유효하다면 제안 프로토콜은 중간자공격에 안전하다.

증명: 위협 상황은 공격자가 각 노드의 nps만 모르는 상황을 가정한다. 각 노드가 공개 채널로 전송하는 (ID, rP, tP, nsP)를 공격자가 가로채고 메시지 변조를 성공했다고 가정한다. 그리고 공격자에 의해 두 개의 세션키가 생성되어 중간자공격에 성공하였다고 가정한다. 하나는 노드 i와 공격자 a 사이의 세션키이다. 다른 하나는 노드 j와 공격자 a 사이의 세션키이다. 여기서 노드 i와 공격자 사이의 세션키만 나타내면 다음과 같다. 노드 i가 계산한 식은 $P_a = (rP_a + H_1(ID_j, H_0(rP_a, nsP_a)))MP$, $K_1 = (t_i + nps_i)(tP_a + P_a)$, $K_2 = (t_i + ns_i)(tP_a + nsP_a)$, $K_3 = (ns_i + nps_i)(nsP_a + P_a)$이다. 공격자가 공격에 성공했으므로 위의 결과와 동일한 K_1과 K_3값을 계산할 수 있다. 2.2절에서 제시한 공격방식을 적용하면 식 (6) 또는 (8) 중 최소 하나는 nps 소거를 통해 노드 i와 동일한 값을 갖도록 변조가 가능하다. 그러나 동시에 두 개의 식을 변조하는 것은 불가능하므로 나머지 하나의 식의 결과 값을 일치시키기 위해서는 nps를 계산하여야 하고 이는 기존 논문에서 증명했듯이 ECDLP에 종속된다. 만일 두 식을 모두 변조 했다고 가정하면 이는 공격자가 $xP = P_a$이 성립하는 x를 빠른 시간 안에 찾았다는 것이며 이는 ECDLP를 해결했다는 뜻이다. 따라서 제안한 프로토콜은 ECDLP가 유효하다면 중간자공격(MITM)에 안전하다.

표 1. 기존 프로토콜과 비교
Table 1. Comparisons with Other Protocols

Protocols	MITM	Computational Costs
He et al.[1]	Vulnerable	$5m+3a+2h$
Farouk et al.[2]	Vulnerable	$3m+3a+2h$
Proposed Protocol	Secure	$5m+4a+3h$

기존 프로토콜과 비교한 결과는 표1과 같다. 세션키 생성 단계에서 제안 프로토콜은 5번의 포인트 곱셈, 4번의 포인트 덧셈, 3번의 해시 함수 호출을 필요로 한다. 여기서 m, a, h는 포인트 곱셈, 덧셈, 해시 함수의 실행 시간이다. 기존 프로토콜과 비교하여 연산 횟수는 다소 증가했지만 제안 방식은 MITM 공격에 안전하다.

Ⅳ. 결 론

본 논문에서는 중간자공격에 안전한 무인증서 기반 인증 키 합의 프로토콜을 제안하였다. 제안 프로토콜은 타원 곡선 이산 대수 문제(ECDLP)가 유효하다면 중간자공격에 안전함을 보였다.

References

[1] D. He, S. Padhye, and J. Chen, “An efficient certificateless two-party authenticated key agreement protocol,” *Computers & Mathematics with Applications*, vol. 64, no. 6, pp. 1914-1926, Sept. 2012.

[2] A. Farouk and M. M. Fouad, “Provable secure pairing-free certificate-less authenticated key agreement protocol,” *2016 4th Int. Conf. Future IoTs and Cloud*, pp. 91-97, Vienna, Austria, October. 2016.