韓國 컴퓨터情報學會 論文誌
第13卷 第5號, 2008. 9.

2008-13-5-4-3

MANET에서 제 3 신뢰기관(TTP)과 사전 키 분배가 필요 없는 강인한 키 교환 방식

한 승 진*, 최 준 혁**

A Robust Pair-wise Key Agreement Scheme without Trusted Third Party and Pre-distributing Keys for MANET Environments

Han Seungjin*, Choi Junhyeog**

요 약

본 논문에서는 유비쿼터스 환경에서 제 3 신뢰기관과 사전 키 분배가 이루어지지 않은 상태에서 노드간 키를 안전하게 교환할 수 있는 방법에 대해서 제안한다. 기존의 방식들은 유비쿼터스 혹은 MANET 환경에서 제 3 신뢰기관(TTP)을 가상적으로 존재한다고 가정하거나, 이미 노드간 암호화 키가 기 배포되었다는 가정에서 연구되었다. 그러나 이러한 방식들은 기반 구조가 없는 무선의 환경에서는 적절치 못한 방법이다. 이런 문제점을 해결하고자 사용되는 방법 중 하나가 Diffie-Hellman 방식을 이용한 방법이다. 그러나 기존의 방법은 중간자 공격과 재생공격에 취약점을 보인다. 따라서, 본 논문에서는 μTESLA 방식을 이용하여 노드들간의 인증문제를 해결하고, 타임스탬프를 이용한 일회용 패스워드 기능을 추가하여 유비쿼터스 환경에서도 안전하고 가벼운 노드간 키 교환 방법을 제안하고 이에 대한 안전성을 검증한다.

Abstract

In this paper, we proposed scheme that it safely exchange encrypted keys without Trust Third Party (TTP) and Pre-distributing keys in ubiquitous environments. Existing paper assume that exist a TTP or already pre-distributed encrypted keys between nodes. However, there methods are not sufficient for wireless environments without infrastructure. Some existing paper try to use the Diffie-Hellman algorithm for the problem, but it is vulnerable to Replay and Man-in-the-middle attack from the malicious nodes. Therefore, Authentication problem between nodes is solved by modified the Diffie-Hellman algorithm using μTESLA. We propose safe, lightweight, and robust pair-wise agreement algorithm adding One Time Password (OTP) using timestamp to modified the Diffie-Hellman in ubiquitous environments, and verify a safety about proposed algorithm.

▸ Keyword : MANET(Mobile Adhoc Networks), Diffie-Hellman, 중간자공격(Man-in-the-middle Attack), 재생공격(Replay Attack), 일회용 패스워드(One Time Password), 키교환(Pair-wise Key)

• 제1저자 : 한승진
• 접수일 : 2008. 6. 10, 심사일 : 2008. 8. 9, 심사완료일 : 2008. 9. 25.
* 경인여자대학 정보미디어학부 조교수 ** 김포대학 e-비즈니스과 부교수

Ⅰ. 서 론

MANET(Mobile Ad hoc Networks)은 고정된 기반 시설(Infrastructure) 없이 이동 노드간 패킷을 주고 받는 무선 네트워크를 의미한다. MANET에서의 이동 노드는 패킷을 송수신하는 노드의 역할 뿐만 아니라, 다른 노드에서 전송되어 온 패킷을 목적지 지역에 연결된 또 다른 노드로 전달해야 하는 라우터 기능까지 포함한다. 따라서 소스 노드로부터 목적지 노드까지의 거리가 1 홉으로 이루어지는 경우도 있지만 대부분 n 홉($n \geq 2$)으로 이루어져 있다. 소스 노드와 목적지 노드의 중간에 악의의 목적을 가진 노드가 존재한다면 많은 문제가 발생한다.

MANET은 중간의 노드를 경유하여 목적지까지 패킷을 전송하기 때문에 중간 노드가 악의의 의도를 갖게 되는 경우 이 노드로부터 패킷을 보호할 수 있는 장치 혹은 방법이 필요하다. 그러나 MANET은 무선 환경에서 기반 시설 없이 오직 노드간 패킷을 주고 받는 방식이기 때문에 보안에 대한 많은 문제점이 드러나고 있다[1-9]. 이에 대한 해결 방법의 하나로 소스 노드와 목적지 노드간 주고 받는 패킷을 암호화하는 방법이 있다. 유선 망에서는 두 노드가 신뢰하는 Trusted Third Party (TTP)를 통해 인증을 받고, 암호 키를 전달 받는다. 그러나 MANET 환경에서는 신뢰할 만한 TTP를 둔다는 것이 MANET 구조상 불가능하다. 또한 모든 노드들에게 사전에 암호화 키를 분배한다는 것도 쉬운 일은 아니다[5].

본 논문에서는 MANET 환경에서 Diffie-Hellman의 키 교환 알고리즘[10]을 이용하여 별도의 TTP를 두지 않으면서, 사전에 노드들에게 키를 배포하지 않고도 노드 간에 안전하게 키(pair-wise)를 주고 받는 방법을 제안한다. 또한 Diffie-Hellman을 이용한 키 교환 알고리즘에서 취약점을 이용한 공격에 대해서는 μTESLA와 암호화된 타임스탬프를 이용하는 One-Time Password (OTP)를 이용하지만, 소스 노드와 목적지 노드 사이에 존재하는 중간 노드들의 타임스탬프를 이용하여 노드간 시간 오차를 최소화하며 타임스탬프의 임계치를 이용하여 OTP의 강인함을 높일 수 있는 키 분배 및 인증 프로토콜을 제안한다. 이와 같은 방법은 MANET에서 뿐만 아니라 유비쿼터스 센서 네트워킹 (USN)에서 센서 노드들도 사용이 가능한 암호화 방법으로서 노드들 간의 안전한 패킷 송수신이 필요한 모든 분야에서 적용이 가능하다.

Ⅱ. 관련 연구

MANET에서 보안에 관한 많은 선행 연구들이 있어 왔다. 기존의 연구들은 크게 안전한 라우팅을 위한 보안(Secure Routing), 악의의 목적을 갖는 노드의 침입 탐지(Intrusion Detection), 키 관리(Key Management), 이기적 노드(Selfish Node)의 탐지 및 관리 등이 있다. 2장에서는 본 논문과 관련있는 키 관리(Key Management)에 대한 기존 연구를 살펴본다.

[1,2]에서는 키 쌍을 교환하기 위해 Diffie-Hellman 방법을 이용하였고, Diffie-Hellman의 취약점을 보완하기 위해 난수를 이용한 OTP를 이용하였다. 그러나 [2]는 유선망 환경에서 동작하는 방법이다. [1]은 자신의 인증을 위해 TESLA를 이용하였고, OTP 사용 시 패스워드의 freshness를 위해 난수를 이용하였다. 그러나 TESLA 방식은 MANET 방식에 적용하기에는 노드들에 부가되는 오버헤드가 크다는 단점이 있고, [8]에서 언급한 것처럼 난수를 이용한 OTP 방식은 이전 세션 키를 이용한 재생 공격(Replay Attack)에 취약하다.

[6,7]에서는 (k, n) 임계치 방식을 이용하여 인증기관의 비밀키를 모든 노드에게 분배하는데, 이 경우 n은 네트워크의 모든 수를 가리킨다. 부분적으로 인증기관의 권한을 분배하는 방식을 개선하기 위해 제안된 이 방법은 네트워크의 모든 노드에게 인증기관의 권한을 분배한다. 이 방법에는 증명서 폐지에 관한 연구가 포함되어 있기 때문에 좀 더 안전한 키 관리 서비스를 제공할 수 있다. 그러나 각각의 노드들이 이웃 노드들의 행동을 감시한다는 가정은 특정 MANET의 경우 매우 큰 오버헤드를 요구할 수도 있다.

[9]에서는 Diffie-Hellman 방식을 사전 연산이 가능한 세션 키와 복잡한 암호 프로토콜에 적용할 수 있도록 일부 변형하여 기존 방식과 강인함은 유사하면서 가벼운 암호 방식을 제안하였다.

Ⅲ. MANET 환경에서 Diffie-Hellman을 이용한 키 교환 시스템의 고찰 및 문제점

MANET은 구조 특성상 기반구조(Infrastructure)가 없기 때문에 TTP를 설치하기가 상당히 어렵다. 또한 모든 노드들은 배터리를 사용한 이동성이 강한 장치(예를 들어, 배터리를 이용한 노트북)이기 때문에 유선 장치에 비해서 전력 공급이 빈약하고, 대역폭이 작으며, 서비스 품질(QoS)이 현저하게 낮다. 대칭 키 방식을 이용하기에는 모든 이동 노드들이

$O(n^2)$의 키를 관리하기란 사실상 불가능하다. 또한 유선망에서의 공개키 방식은 모든 노드들이 $n-1(n \geq 2)$개의 공개키를 모두 관리하여야 하므로 노드들에게 상당한 오버헤드로 작용한다. 따라서 데이터를 전송하기 전에 암호화 키를 생성하기 위해 TTP를 통하지 않고, 또한 미리 분배되지 않은 상황에서 상대 노드와 암호화 키를 교환하여 이를 이용하여 패킷을 암호화하는 것이 MANET 환경에서 효율적이며 현실적이다.

Diffie-Hellman의 키 교환 알고리즘은 사전의 키 분배가 없고, TTP가 없이 두 노드간에 키 교환이 가능하다. 이는 MANET과 같이 기반 구조가 없는 환경에서 키 교환 방식으로 적합하다. 그러나 기존의 Diffie-Hellman 키 교환 알고리즘을 MANET에 적용하기에는 중간자 공격(Man-in-the-middle Attack)과 재생 공격(Replay Attack)[1,2,8]에 취약하다.

본 논문의 키 교환 시스템은 Diffie-Hellman과 타임 스탬프를 이용한 OTP를 이용하여 TTP와 사전의 키 분배 없이 노드간 안전하게 키 교환을 할 수 있다. 이것이 가능하려면 본 논문의 모델 즉, MANET 환경에 적용하는 Diffie-Hellman의 문제점을 파악하고 이를 해결하여야 한다. 또한 Diffie-Hellman의 취약점을 보완하기 위해 사용하는 기존의 OTP의 문제점 역시 해결해야 한다. 따라서 본 논문은 기존의 Diffie-Hellman을 이용한 키 교환 시스템의 문제점을 해결한 강인한 키 교환 시스템을 설계하고, 이에 대한 안전성과 강인성에 대해 검증한다.

3.1 용어 정의

다음은 본 논문에서 사용하는 용어들의 정의이다.

A, B : 사용자
M : 악의의(Malicious) 노드
x_0^A : 노드 A의 비밀 메시지(혹은 패스워드)
a : 노드 A의 개인 키
$g_a \leftarrow g^a \pmod p$: 노드 A의 공개 키
k_n : n 번째 세션 키
$H(\ )$: 단방향 해쉬 함수
$H(g_a)$: 노드 A의 공개 키 검사
$x_k^A \leftarrow H^k(x_0^A)$: $H(\ )$를 이용하여 x_0^A 를 k 번 해쉬한 결과 값
$c \leftarrow E_K\{d\}$: K 를 이용하여 평문 d 를 암호문 c 로 대칭적으로 암호화
$d \leftarrow D_K\{c\}$: K 를 이용하여 암호문 c 를 평문 d 로 대칭적으로 복호화
g : 곱셈 군(multiplicative group) Z_p^*의 생성자(generator), p의 원시근(primitive root)
p, q : 강한 소수(strong prime), p = 2 × q + 1
K' : 소스 노드와 목적지 노드가 이전 세션에서 사용하던 키
K_{AB} : 노드 A와 노드 B가 공유하는 비밀 키
N_B : 노드 B에서 전송한 난수
T_S : 소스 노드의 타임스탬프
T_K : K 번째 노드의 타임스탬프
T_D : 목적지 노드의 타임스탬프
$T_{N_{SD}}$: 소스 노드와 목적지 노드의 세션 종료 시간
$T_{threshold}$: 각 노드가 수용하는 타임스탬프 임계치

3.2 중간자 공격

MANET의 구조 특성상 소스 노드에서 목적지 노드까지가 1-홉인 경우도 있지만 대부분 2-홉 이상으로 소스 노드와 목적지 노드 사이에 중간 노드들이 존재한다. 따라서 소스 노드와 목적지까지의 거리가 2-홉 이상인 경우는 중간 노드를 경유해야 패킷을 전달할 수 있다. 이때 중간 노드는 소스 노드에게 자신이 목적지 노드라고 위장하고, 목적지 노드에게는 자신이 소스 노드라고 위장하면, 소스 노드와 목적지 노드만이 공유해야 하는 키를 중간 노드가 알 수 있다. Diffie-Hellman 키 교환 방식은 소스 노드와 목적지 노드가 상호 인증(Mutual Authentication)을 하지 않기 때문에 [그림 1]처럼 중간자 공격이 가능하다. 다음은 중간자 공격을 나타낸 것이다.

① 노드 A는 $a \in_A [1, p-1]$을 생성하고, $g_a \leftarrow g^a \pmod p$를 계산한다. 노드 A는 g_a를 노드 M(노드 B)에게 전송한다.

② 노드 M은 $m \in_M [1, p-1]$를 선택해서 $g_m \leftarrow g^m \pmod p$를 계산한다. 노드 M은 g_m을 노드 B에게 전송한다.

③ 노드 B는 $b \in_B [1, p-1]$을 생성하고, $g_b \leftarrow g^b \pmod p$를 계산한다. 노드 B는 g_b를 노드 M(노드 A)에게 전송한다.

④ 노드 M은 노드 A에게 ②와 같은 방법을 이용하여 g_m을 생성한 후 전송한다.

* 노드 A는 $k_1 \leftarrow g_m^a \pmod p$를 계산한다.

(이때 k_1은 노드 M이 $k_1 \leftarrow g_a^m \pmod p$를 계산할 수 있기 때문에 노드 A와 노드 M이 공유하는 것이 가능하다.)

* 노드 B는 $k_2 \leftarrow g_m^b \pmod p$를 계산한다.

(이때 k_2는 노드 M이 $k_2 \leftarrow g_m^b \pmod p$를 계산할 수 있기 때문에 노드 M과 노드 B가 공유하는 것이 가능하다.)

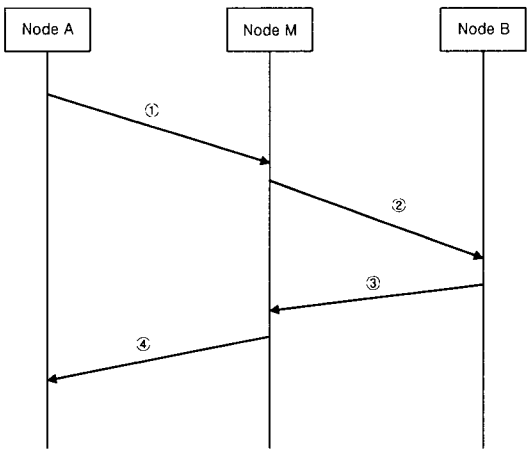

그림 1. 중간자 공격
Fig 1. Man-in-the-middle Attack

Diffie-Hellman을 사용하기 위해서는 중간자 공격 때문에 키 교환을 하기 전에 노드간 상호인증을 하여야 한다. [3]과 같은 방식은 유선망에서 인증 서버(CA)를 두고 인증 장치를 통해 상호 인증이 가능하다. 그러나, MANET에서 신뢰할 수 있는 TTP를 둔다는 것은 불가능하기 때문에 TTP 없이 상호 인증이 가능한 μTESLA[11] 방식을 적용한다. [1]에서는 노드들 간의 상호 인증 시 TESLA를 이용하였지만, TESLA 기술은 초기 패킷에 전자서명을 포함하여 인증하는 방식으로 시스템의 부하를 가중시킨다. μTESLA에서는 대칭 메커니즘만 사용하여 인증 절차를 수행한다. 또한 TESLA 방식은 각 패킷에 사용된 키를 노출하여 송수신함으로써 많은 에너지의 소모가 발생한다. 그러나 μTESLA에서는 주기별로 한 번씩만 키를 노출시켜 자원 효율을 극대화시킨다.

μTESLA는 TESLA를 센서 노드에 적합하도록 인증부분만 분리한 것으로 인증 키 체인 생성이나 브로드캐스트 데이터 생성 방식은 TESLA와 유사하다. μTESLA가 베이스 스테이션만이 센서 노드들에게 브로드캐스트할 수 있는 것을 본 논문에서는 모든 노드들이 가능하도록 수정한다. 그러나 μTESLA는 모든 노드들이 시간 동기화가 되어 있어야 하고 실제 네트워크 전송 지연이 있어 키 노출 지연 시간의 설정이 필요하다.

그리고, 모든 노드들이 시간 동기화를 하지 않고, 패킷을 송수신하는 노드들끼리만 동기화를 한다. 시간 동기화를 위해서는 노드들 간의 타임스탬프를 이용하지만, 모든 노드들의 모든 시계가 모두 완전하게 일치되는 것을 기대할 수 없으므로 적당한 시간오차(Clock Skew)를 고려한다. 또한, 시간오차를 최소화하고, 다음에서 설명하는 재생 공격을 방지하기 위해 소스 노드와 목적지 노드를 포함한 경로에 있는 노드들의 타임스탬프를 이용한다.

노드들 간의 시간 동기화 문제는 [그림 3]처럼 소스 노드가 목적지 노드로 패킷을 전송할 때 패킷에 타임스탬프를 포함해서 전송한다. 이를 수신한 노드 A는 암호화된 자신의 타임스탬프를 패킷에 추가하여 노드 B에게 전달한다. 노드 B는 노드 A와 같이 암호화된 타임스탬프를 추가하여 다음 노드로 전송하고, 최종적으로 목적지 노드는 패킷을 수신한다. 목적지 노드의 타임스탬프(T_5)는 소스 노드의 타임스탬프(T_1)와 중간 노드들의 타임스탬프(T_2, T_3, T_4)를 참조하여 소스 노드와의 시간을 동기화한다.

3.3 재생 공격

재생 공격이란, [그림 2]처럼 A가 B에게 보내는 메시지가 있을 때 중간 위치한 M이 A와 B의 세션이 종료된 후 이전에 사용되던 세션 키를 이용하여 마치 A(혹은 B)처럼 위장한 후 B(혹은 A)와 메시지를 주고 받는 공격방법이다.

① $\{A \rightarrow M(B) : K'\}_{K_{AB}}$

노드 A는 노드 M(B)에게 A와 B가 공유하는 비밀키 K_{AB}를 이용하여 세션키 K'을 전송한다.

② $\{M(A) \rightarrow B : K'\}_{K_{AB}}$

노드 M(A)은 노드 B에게 A와 B가 공유하는 비밀키 K_{AB}를 이용하여 암호화된 세션키 K'을 전달한다.

③ $\{B \rightarrow M(A) : N_B\}_{K'}$

노드 B는 세션 키 K'를 이용하여 난수 N_B를 암호화하여 노드 M(A)에게 전송한다.

④ $\{M(B) \rightarrow A : N_B\}_{K'}$

노드 M(B)는 세션 키 K'를 이용하여 암호화된 난수 N_B를 노드 A에게 전달한다.

⑤ $\{A \rightarrow M(B) : N_B - 1\}_{K'}$

노드 A는 노드 M(B)에게 세션 키 K'를 이용하여 난수

N_B-1를 암호화하여 전송한다.

⑥ $\{M(A)\rightarrow B: N_B-1\}_{K'}$

노드 M(A)은 노드 B에게 세션 키 K'를 이용하여 암호화된 난수 N_B-1를 전달한다.

세션이 종료된다.

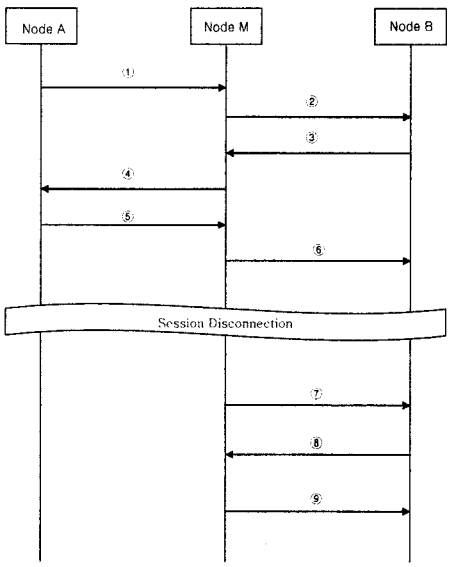

그림 2. 재생 공격
Fig 2. Replay Attack

⑦ $\{M(A)\rightarrow B: K'\}_{K_{AB}}$

노드 M(A)는 노드 B에게 이전에 노드 A와 노드 B와의 연결에서 이용되었던 비밀키 K_{AB}와 노드 A와 노드 B와의 이전 세션에서 사용되었던 세션키 K'을 전송한다.

⑧ $\{B\rightarrow M(A): N_B\}_{K'}$

노드 B는 세션 키 K'를 이용하여 난수 N_B를 암호화하여 노드 M(A)에게 전송한다.

⑨ $\{M(A)\rightarrow B: N_B-1\}_{K'}$

노드 M(A)는 노드 B에게 이전 연결에서 사용되었던 세션 키 K'를 이용하여 암호화된 난수 N_B-1를 전달한다. 노드 B는 노드 M을 노드 A로 오인하여 세션이 연결된다.

위와 같은 재생 공격에 대해 [1,2]는 난수를 이용한 OTP[4]를 사용하였다. OTP는 제 3의 악의 노드가 재생 공격을 하더라도 각 연결시마다 새로운 난수와 키 체인 요소가 생성되기 때문에 재생 공격이 불가능하다. 그러나 [그림 2]에서 언급한 것처럼 난수를 이용한 OTP는 중간의 악의의 노드가 이전 세션에서 난수를 이용한 세션 키를 저장하고 있다가, 다음 세션에서 해당 세션 키를 이용하면 연결이 가능하다[8].

본 논문에서는 각 노드들의 인증에 μTESLA를 사용함에 따라 필요한 노드들의 시간 동기화 문제를 해결하고, OTP에서 패스워드의 freshness를 위해 사용하는 난수 대신에 타임스탬프를 사용한다. 노드들 간의 시간 오차(Clock Skew; 커버로스[3]에서는 5분을 시간오차로 두고 있다)를 극복하기 위해 소스 노드는 목적지 노드에게 타임스탬프를 전송한다. 소스 노드의 타임스탬프가 추가된 메시지를 수신한 중간 노드는 소스 노드에게서 자신까지 패킷이 전송되는데 소요된 시간 정보를 추가하여 다음 노드로 전송한다. 이를 수신한 다음 노드 역시 시간 정보를 추가하여 다음 노드로 전송한다. 이러한 과정을 거친 후 타임스탬프가 추가된 메시지가 목적지 노드에게 최종적으로 전달이 되면 목적지 노드는 소스 노드가 전달한 타임스탬프와 중간 노드들이 추가한 타임스탬프를 이용하여 자신의 시간 정보와 비교한다. 이때 특정 노드가 추가한 타임스탬프가 이전 노드보다 작거나 현저하게 크다면 목적지 노드는 이를 악의의 노드로 간주하고 해당 타임스탬프를 삭제한다(악의의 노드 검출 방법은 본 논문의 범위에서 벗어나므로 제외한다). 시간 차이가 식 (3.1)의 조건을 만족하면 이를 무시하고, (3.1)의 조건을 만족하지 못하면 시간 정보를 수정한다.

$$|Clock-T| < \Delta t_1 + \Delta t_2 \quad \cdots\cdots (3.1)$$

여기서, T는 송신 노드가 전송한 타임스탬프이고, Clock은 수신자의 로컬 타임스탬프이고, Δt_1은 소스 노드와 목적지 노드간의 시간 차이이다. Δt_2는 노드 간 패킷 전달 지연 시간이다.

본 논문에서는 기존의 OTP를 이용하지만 전송하는 메시지에 난수 정보대신에 타임스탬프 정보를 이용한다. 여기서, Δt_1과 Δt_2를 사용하지 않고 타임스탬프를 이용하는 이유는 특정 노드(악의의 목적을 갖는 노드)가 이전 노드에게서 수신한 타임스탬프로부터 정확한 시간정보를 추가하지 않고, 큰 Δt_2 값을 추가함으로써 이전에 사용되었던 세션 키를 사용할 수 있도록 하는 것을 방지하기 위함이다.

목적지 노드는 소스 노드와 중간 노드들이 전송한 타임스탬프를 이용하여 자신의 시간을 동기화하고(상호 인증을 하기 위해 μTESLA를 사용하기 위한 방법도 포함) 소스 노드와의 시간 오차를 최소화하여 재생 공격으로부터 대비한다.

IV. TTP와 사전 키 분배가 필요없는 키 교환 시스템

4.1 Diffie-Hellman과 타임스탬프 OTP를 이용한 강인한 키 교환 시스템

이장에서는 3.2와 3.3에서 설명한 것처럼 기존의 Diffie-Hellman의 단점을 보완한 강인한 키를 교환하는 방법에 대해서 설명한다.

소스 노드는 자신의 인증 검사에 대한 무결성을 위해 μTESLA를 이용한다. 이때 μTESLA에서 필요한 노드들 간의 시간 동기화는 소스 노드와 목적지 노드간 주고 받는 메시지내의 타임스탬프를 이용하여 동기화 한다. 여기서는 센서 노드에서 적용되는 μTESLA처럼 모든 노드들 간의 시간 동기화가 필요 없고, 오직 소스 노드와 목적지 노드간만이 동기화가 필요하다.

본 논문에서는 기존의 Diffie-Hellman 방법을 다음과 같이 수정하여 강인한 키 교환 시스템을 제안한다.

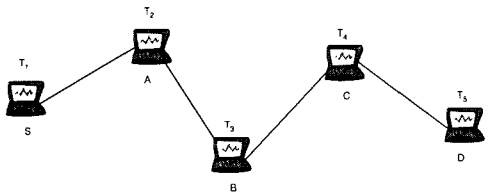

그림 3. 수정된 Diffie-Hellman을 적용한 네트워크 구조
Fig. 3 A Network structure adopted modifying Diffie-Hellman scheme

소스 노드(S)$_j$는 $a \in {}_A[1, p-1]$를 선택하고, g^a 를 계산한다.

$$\left\{S \rightarrow A(D): ID_A, E_{x^A_{n_A - i}}\left(g^a, T_S\right), x^A_{n_A - i + 1}, i\right\} \quad \cdots\cdots (4.1)$$
$$where, 1 \le (n_A, i) \le k$$

[그림 3]처럼 메시지 (4.1)을 수신한 노드 A는 자신이 암호화한 타임스탬프를 추가하여 다음 노드인 노드 B로 메시지를 전달한다.

$$\left\{A \rightarrow B(D): [ID_A, E_{x^A_{n_A - i + 1}}\left(g^a, T_S\right), x^A_{n_A - i + 2}, i], T_2\right\} \quad (4.2)$$

:

$$\left\{C \rightarrow D: [[[ID_A, E_{x^A_{n_A - i + (k-1)}}\left(g^a, T_S\right), x^A_{n_A - i + (k)}, i], T_2], \ldots, T_{k-1}\right\}$$
$$\cdots\cdots (4.3)$$

목적지 노드 D는 다음 (4.4)와 (4.5)를 체크한다.

$$H^k(x^A_{n_A - i + k}) = x^A_{n_A} \quad \cdots\cdots (4.4)$$
$$T_S < T_2 < \ldots < T_{k-1} < \ldots < T_D \quad \cdots\cdots (4.5)$$

소스 노드 S가 목적지 노드 D가 이전에 메시지를 송수신하여 [표 1]과 같은 테이블에 존재한다면 소스 노드 S의 타임스탬프 T_S는 $T_{N_{SD}}$보다 커야 한다. 즉, (4.6)과 같은 조건을 만족해야 한다.

$$T_S > T_{N_{SD}} \quad \cdots\cdots (4.6)$$

여기서, $T_{N_{SD}}$는 소스 노드 S와 목적지 노드 D의 세션이 종료된 시간이다.

여기서 특정 노드(N)의 타임스탬프의 값이 다음 노드(N+1)의 타임스탬프 값보다 작거나, 이전 노드들의 타임스탬프에 비해서 현저히 크다면 목적지 노드는 해당 노드를 공격자로 가정한다.

이때, 소스 노드 S와 목적지 노드 D와의 세션 종료시간이 [표 1]에 존재하지 않는다면 목적지 노드 D는 소스 노드 S가 최초 전송으로 간주한다. 그러나 $T_{N_{SD}}$는 다음을 만족해야 한다.

$$T_{threshold} < T_{N_{SD}} \quad \cdots\cdots (4.7)$$

여기서, $T_{threshold}$는 각 노드가 정한 다른 노드들과의 세션 종료 시간에 대한 임계치이다.

공격자는 키의 freshness를 위해 가급적 최근에 종료된 세션 키를 이용하려 할 것이다. 따라서 $T_{N_{SD}}$가 임계치 $T_{threshold}$ 범위를 벗어난다면, 목적지 노드는 재생 공격으로 간주하고 해당 메시지를 폐기한다. 목적지 노드는 해당 메시지를 폐기한 후 이 메시지를 전송한 노드를 악의의 노드로 결정할지 여부와 이웃 노드로 알리는 방법은 본 논문의 범위에서 벗어나므로 생략한다. 따라서, 적절한 $T_{threshold}$의 추후 연구가 필요하다.

각 노드는 다음과 같은 테이블을 생성하고 관리한다.

〔표 1〕 연결이 종료된 노드들과의 세션 종료 시간
〔Table 1〕 Session end time with disconnected neighbor nodes

노드	세션 종료 시간
1	h
2	j
3	k
:	:
N	x

여기서, h, j, k, ... , x는 각각 노드 1, 2, 3, ... , N 이 〔표 1〕을 관리하는 노드와 세션이 종료된 시간이다. 각 레코드는 일정 시간이 지나면 삭제된다.

목적지 노드는 위의 조건이 만족한다면 $x^A_{n_A-i+(k-1)}\{g^r\}$과 $x^A_{n_A-i+k}$를 저장한다.

목적지 노드(D)는

$b \in {}_A[1, q-1]$를 선택하고, g^b 를 계산한다.

나머지는 소스 노드가 목적지 노드로 전송하는 (4.1) ~ (4.7)과 유사하다.

기존 연구에서 살펴본 것처럼 Diffie-Hellman 방법은 중간자 공격과 재생 공격에 취약하다. 본 논문은 Diffie-Hellman을 이용하여 키 교환 시 중간자 공격을 위해서는 μTESLA 방식을 이용하여 상대 노드에게 자기 인증을 한다. 그러나 μTESLA 방식은 TESLA 방식에 비해 MANET 환경에는 적합하지만 노드들 간의 시간 동기화가 문제가 된다. 재생 공격을 위해서는 기존 연구처럼 OTP를 이용하지만 난수를 이용한 OTP 방식은 중간의 악의의 노드에 의해 재생 공격이 가능하다.

따라서, 본 논문에서는 노드들의 암호화된 타임스탬프를 이용하여 노드들 자신의 인증에 사용하는 μTESLA의 시간 동기화 문제를 해결하고, 이러한 암호화된 타임스탬프는 재생 공격 방지를 위해 사용된다. 따라서 중간자 공격을 위해 목적지 노드에서는 수신한 메시지의 무결성과 기밀성을 보장받도록 한다.

또한 Diffie-Hellman 방법을 이용하여 long-term 키 중 개인키를 생성하고 공개키를 계산하는 사용자 그룹(임의의 사용자 A, B)이 있다고 가정하면 이들에 대한 공개 값, 즉 q와 g에 대한 전체 공개 값 모두는 임의의 제 3 신뢰기관(TTP)에 저장된다. 이와 같이 long-term {공개 | 개인} 키는 신뢰할 수 있는 TTP가 존재할 경우 높은 신뢰성과 높은 인증 정도를 제공할 수 있다. 오로지 A와 B만이 TTP에 접근하여 키를 정할 수 있기 때문이다. 그러나 본 논문은 MANET 환경에서의 연구이므로 TTP의 부재와 강인한 보안성을 위해 long-term 키는 사용하지 않고, 개인 키와 공개 키는 암호 키를 전달할 때마다 생성하여 사용한다.

4.2 분석

본 논문에서 연구하는 시스템의 보안성을 평가하기 위해 다음과 같은 〔5〕에서 제시하는 요구사항 중 기존의 Diffie-Hellman 알고리즘의 취약점으로 알려져 온 재생 공격과 중간자 공격에 대해 만족하는지 분석한다.

도청 공격(Eavesdropping) : 공격자는 Diffie-Hellman 알고리즘의 특성상 암호화 키 K^+와 복호화 키 K^-를 알 수 없기 때문에 도청 공격이 불가능하다.

재생 공격(Replay attack) : 각 연결 시 OTP를 사용하기 때문에 공격자는 재생 공격이 불가능하다. 본 논문에서는 3.3과 3.4절에서 보인 것처럼 기존의 연구에서 사용한 난수를 이용한 OTP의 문제점을 보이고, 이에 대한 해결책으로서 소스 노드와 목적지 노드간의 타임스탬프를 이용하여 클럭을 동기화하여 보다 안전한 OTP를 사용하기 때문에 재생 공격이 불가능하다.

중간자 공격(Man-in-the-Middle) : 중간자 공격은 소스 노드와 목적지 노드간 상호 인증을 하지 않기 때문에 발생한다. 본 논문에서는 이동 노드들의 특성 상 노드간 상호 인증을 위해 μTESLA를 이용하고, 이에 대한 Clock Skew 문제를 해결하기 위해 노드들간의 타임스탬프 값을 암호화 하여 노드간 전달하도록 한다. 이를 이용하여 소스 노드와 목적 노드 사이에 존재하는 공격자는 소스 노드와 이전 노드의 키 값을 알지 못하기 때문에 타임스탬프 값을 위변조하지 못하기 때문에 본 논문의 방식은 중간자 공격으로부터 안전하다.

V. 결론 및 향후 과제

본 논문에서는 유비쿼터스 환경으로 진입하기 위한 핵심기술 중의 하나인 MANET에서 소스 노드와 목적 노드간에 TTP와 사전 키 분배 없이 안전하게 암호화 키를 교환하는 방법에 대해서 제안하고, 분석을 통해 기존의 Diffie-Hellman 알고리즘의 문제점인 재생 공격과 중간자 공격에 대해 안전함

을 입증하였다.

본 논문의 결과를 이용하여 MANET 환경에서 TTP가 없으면서 사전에 키 분배 없이 안전하고, 강인한(robust) 키(pair-wise) 교환이 가능하고, 사용자 인증이나 세션 키 공유 및 확인을 요하는 시스템에 유용하게 적용이 가능하다. 또한 각 노드들이 관리하는 세션 종료 테이블을 이용하여 악의의 노드 검출이 가능하며, USN에서 자기 인증을 위해 필요한 μTESLA를 위해 시간 동기화 문제 해결의 한가지로 가능하다. 향후 RFID/USN에서 사용가능한 가벼운(lightweight) 암호키 교환 메커니즘으로의 응용이 가능하다.

참고문헌

[1] G. C. Wang, G. H. Cho, and S. W. Bang, "A Pair-wise Key Establishment Scheme without Pre-distributing Key for Ad-hoc Networks," ICC'05, vol., 5, pp.3520~3524, 16~20, May, 2005.

[2] 서승현, 조태남, 이상호, "OTP-EKE : 원-타임-패스워드 기반의 키 교환 프로토콜," 정보과학회 논문지 : 시스템 및 이론, 제 29권 제 5호, 한국정보과학회, June, 2002.

[3] S. Miller er. al., "Kerberos Authentication and Authorization System," Section E.2.1, Project Athena Technical Plan, MIT. Project Athena, Cambridge, MA. 27 Oct., 1988.

[4] N. M. Haller, "The S/KEY. one-time password system," In Proceedings of the Symposium on Network and Distributed System Security, pp. 151~157, 1994.

[5] S. Zhu, S. Xu, S. Setia, and S. Jajodia, "Establishment pair-wise keys for secure communication in ad hoc networks: a probabilistic approach," In Proceedings of the 11th International Conference on Network Protocols, pp. 326~335, 2003.

[6] J. Kong, P. Zerfos, H.Luo, S. Lu and L. Zhang, "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," ICNP'01, 2001.

[7] H. Luo, J. Kong, P. Zerfos, S. Lu and, L. Zhang, "Self-securing Ad Hoc Wireless Networks," ISCC'02, 2002.

[8] Wenbo Mao, Modern Cryptography : Theory and Practice, Prentice Hall, July, 2003.

[9] 양대헌, 이경희, "변형 Diffie-Hellman 키 교환 프로토콜," 정보처리학회 논문지, 제14-C권, 제6호, 한국정보처리학회, Oct., 2007.

[10] W. Diffie and M. Hellman, "New Directions on Cryptography," IEEE Transactions on Information Theory, IT-22(6):644~654, Nov., 1976.

[11] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," Proc., of the 7th ACM/IEEE International Conference on MobiCom, 2001.

저자소개

한 승 진

1990 인하대학교 전자계산학과 학사
1992 인하대학교 전자계산공학과 석사
2002 인하대학교 전자계산공학과 박사
1992~1996 대우통신 종합연구소
1996~1996 한국전산원 초고속사업단
1996~1998 SKTelecom 디지틀사업본부
2002~2004 인하대학교 컴퓨터공학부 강의조교수
2004~현재 경인여자대학 정보미디어학부 조교수
2007~현재 TTA PG103 표준화위원
관심분야 : USN, MANET, Mobile Computing, 임베디드 시스템, Security

최 준 혁

1990 경기대학교 전자계산학과 학사
1995 인하대학교 전자계산공학과 석사
2000 인하대학교 전자계산공학과 박사
1997~현재 김포대학 e-비즈니스과 부교수
2001~2002 한국전자통신연구원 컴퓨터소프트웨어연구소(초빙연구원)
2003~현재 특허청 특허출원 심사자문위원
2003~현재 김포발전연구소 소장
관심분야: 정보검색, 유전자 알고리즘, 신경망, USN, 임베디드 시스템, 전자상거래 보안