무선 센서네트워크 환경에 효율적인 키 관리 프로토콜

김현욱*, 김태연**
*전북대학교 컴퓨터정보학과
**서남대학교 컴퓨터정보통신학과
e-mail: hwkim@dcs.chonbuk.ac.kr, tykim@seonam.ac.kr

An Efficient Key Management Protocol for Wireless Sensor Network Environment

Hyoun-Wook Kim*, Tae-Yeon Kim**
*Dept of Computer Information, Chon-Buk University
**Dept of Computer Science and Information Communications, Seonam University

요 약

무선 센서네트워크 환경에서 대칭형 암호화 알고리즘을 적용하는 경우에 제한된 자원과 키 길이로 인하여 다양한 보안 공격에 취약하다. 이러한 문제해결을 위해 온라인상에서 중앙 키 관리 센터를 통해 정기적으로 사용 중인 키를 갱신하는 메커니즘이 필요하다. 여기서 제안된 프로토콜은 정기적인 키 갱신 및 키 정보 인증을 실시하며, 송신자는 키 정보를 암호화하지 않고 공개된 하나의 랜덤 비트 패턴으로 방송한다. 또한 간단하고 빠른 비트위주 XOR 연산을 사용한다. 제안된 키 관리 프로토콜이 네트워크의 성능 측면에서 다른 프로토콜보다 우수함을 보이기 위해 기존의 대칭키 암호화 프로토콜, 마스터 키 기반 SPINS 프로토콜과 비교분석한다.

1. 서론

무선 센서네트워크 (Wireless Sensor Network : WSN)에서 일반적으로 노드들은 제한된 자원(전력, 메모리, 프로세서, 네트워크 대역폭)을 가지기 때문에 소비 전력을 최소화할 수 있는 보안 프로토콜 요구되며, 기존의 유선 네트워크의 비대칭형보다는 보다 간단하고 처리비용이 낮은 대칭형 암호화 알고리즘을 사용하며, 이 경우 센서 노드는 제한된 키 길이와 메모리를 사용하므로 보안 공격에 취약하다. 이에 대한 해결책은 세션키 길이를 확장하거나 세션키 정기적으로 갱신하는 것이다.

본 논문에서는 대칭형 암호화 알고리즘을 기반으로 하는 효율적인 키 관리 프로토콜을 기술한다. 키를 갱신하거나 생성하는 과정에서 발생하는 계산비용을 줄이기 위해 비트 위주 XOR 연산을 사용해서 세션키를 분배하는 메커니즘을 제안한다.

본 논문의 구성은 서론에 이어 2장에서 관련연구를 기술하고, 3장에서 WSN 환경에서 보안 문제를 설명한다. 4장에서 세션키 생성과 갱신 프로토콜을 기술하고, 5장에서는 통신 오버헤드와 각 노드의 계산 비용, 메모리 사용 측면에서 기존의 프로토콜과 비교 및 분석한다. 마지막으로 결론과 향후 연구방향을 제시한다.

2. 관련연구

Carman [2] 등은 WSN에서 키 일치와 분배를 위한 다양한 접근방식을 제안하였다. 그들은 다양한 하드웨어 플랫폼에 대해 각 프로토콜들의 오버헤드 측면에서 비교분석하였다.

Jamshaid [3] 등은 신뢰된 키 서버에 의해 키를 생성되고 분배되는 메커니즘과 기존의 네트워크에 가입(join)을 원하는 새로운 노드나 네트워크로부터 탈퇴(leave)하는 노드가 있는 경우에 그룹키를 갱신하는 프로토콜을 제안하였다. 그러나 이 구조는 키 교환과 인증을 위해 공개키 방식을 사용하는 단점을 가지고 있다.

Eschenauer [4] 등은 키 사전 분배 방식을 제안하였다. 즉, 각 노드들은 지역에 배치되기 전에 키 풀로부터 키 집합을 선택하고 저장한 다음에 배치된 후에 인접 노드와 서로 공유하고 있는 키를 찾아서 비밀키로 사용하는 방식이다. 그러나 이 구조는 1 대 1 통신에는 적합하지만 1 대 n 통신 환경에는 적합하지 않는 방식이다.

Perrig [5] 등은 SPINS에서 BS가 자신과 각 노드 간에 데이터를 안전하게 교환하는데 사용할 세션키와 인증을 위한 MAC 키를 관리하며 SNEP가 두 노드간에 교환되는 데이터의 비밀성을 보장하고 인증을 수행할 수 있는 구조이고, TESLA가 지연된 대칭키의 공개(disclosure) 방식으로 비대칭 원리를 도입한 인증된 멀티캐스트 기능을 제공한다. 그러나 이 구조는 노드가 방송된 데이터를 읽는 것이 일정 시간 후에 가능하기 때문에 실시간 처리 환경에는 부적합하다.

Jolly [6] 등은 WSN을 위한 계층적인 키 관리 구조를 제안하였다. 이 구조는 각 노드들은 자신과 연결된 게이트웨이 간에 키를 공유하는 방식으로 이 중간 노드는 모든 메시지에 대해 암·복호화 처리 과정을 수행해야 하기 때문에 많은 오버헤드를 받게 된다.

Cam [1] 등은 키 갱신 과정에서 세션키를 안전하게 관리하기 위해 다른 키로 암호화하는 방식을 사용하지 않고 계산 속도가 빠른 비트 XOR 연산을 사용하는 방식을 제안하였다. 즉, 센서 노드들이 배치되기 전에 BS와 각 노드만이 공유하는 비밀키를 분배하고 새로운 키로 갱신하고자 하는 경우에 모든 노드들에게 공개된 하나의 비트 스트림을 온라인으로 전송하고, 각 노드는 수신한 비트 스트림을 자신이 관리하고 있는 세션키와 XOR 연산을 통해 새로운 세션키를 생성하는 방식이다. 그러나 그들의 논문에는 BS와 각 노드 간의 1 대 1 통신을 위한 키 관리 프로토콜을 제안했을 뿐 1 대 n의 통신, 노드와 노드간의 통신을 위한 서브 그룹키 관리에 대해서는 언급되어 있지 않다.

3. WSN 보안 요구사항

3.1 요구조건

신뢰성 있는 WSN을 운용할 수 있도록 하기 위해 BS는 고성능의 처리 기능과 전력, 대규모 메모리를 가지고 있으며, 신뢰된 노드로 가정한다. 그러나 각 노드들은 제한된 전력, 계산·통신 능력을 가지며, 노드 간에 무선 링크로 통신으로 보안에 매우 취약하기 때문에 각 노드들은 해당 지역에 배치되기 전에 초기 세션키와 보조키를 받는다고 가정한다.

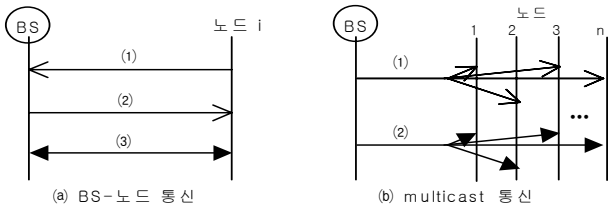
3.2 표기

보안 프로토콜과 암호화 연산을 기술하기 위해 다음과 같은 표기법을 사용한다.

- A, B : 노드의 식별자 A, B
- $S(S_i)$, N_A : 모든 노드들(멤버)과 노드 A에 의해서 생성된 난수
- KI, SK_i, sK_i : 키정 보, BS와 노드 i가 공유하는 세션키와 보조키
- GKs, $K_{A,B}$: 모든 노드가 공유하는 그룹키와, 노드 A와 노드 B만이 공유하는 임시 키
- [M]SK : 키 SK를 사용하여 메시지 M을 암호화한 메시지
- $MAC(SK_i, M)$: 세션키 SK_i를 사용하여 메시지 M에 대한 MAC를 계산한 값
- 단방향 해쉬 체인함수 $F()$: $K_i = F(K_{i+1})$, $0 \le i \le n-1$

4. 세션키 생성과 갱신 프로토콜

4.1 BS와 노드간의 세션키 갱신



(그림 1) BS와 노드, BS와 노드들 간의 통신

BS와 노드간의 통신에 이용중인 세션키를 새로운 세션키로 갱신하는 절차는 다음과 같다(그림 1(a)).

1. BS는 주기적으로 키를 갱신하기 위해 각 노드의 새로운 세션키($SK_i' = KI \oplus SK_i, (i=0,n)$)을 계산하고 자신의 메모리에 저장한다.
2. BS는 일정 기간이 지나면 키를 갱신하기 위해 각 노드에게 메시지(00, null, KI_{BS}', T_s, T_{int})을 방송한다(그림 1(a)메세지(2)). 이러한 메시지의 내용은 모든 센서 노드뿐만 아니라 공격자들에게도 공개된다. 여기에서 00은 개인 세션키의 갱신을 나타내는 필드이고, 두 번째 필드인 null은 특정 노드가 아닌 모든 노드를 나타내며, KI_{BS}'는 BS가 생성한 키 정보로서 $KI_{BS} = F(KI_{BS}')$의 관계를 만족한다. T_s와 T_{int}는 BS가 메시지를 전송한 시각과 세션키의 사용 기간을 나타낸다.
3. 메시지를 수신한 각 노드는 메시지의 인증과 새로운 세션키를 계산한다(그림 1(a)메세지(3)).

① 키정보(KI_{BS}')가 BS에 의해서 보내졌는지를 검사하기 위해 단방향 해쉬 체인함수 F를 적용한다. 즉, 수신 메시지 내에 포함된 KI_{BS}'와 이전에 수신한 KI_{BS}에 대해 조건식($F(KI_{BS}') = KI_{BS}$)이 만족하는지를 검사한다. 서로 일치하면 BS에 의해 생성 정보로 인정한다.

② 각 노드는 세션키($SK_i'=KI\oplus SK_i$)를 생성하고 단계 5로 이동한다.

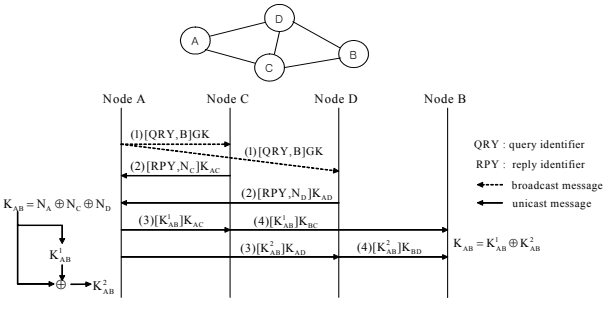
4. 단계 3에서 키정보에 대한 인증이 이루어지지 않으면 오류 메시지를 BS에게 통보하고 수신한 메시지를 폐기한다.
5. BS와 노드는 새로운 세션키(SK_i')에 의해서 암호화된 데이터를 전송한다.

4.2 BS와 노드들 간의 그룹키 갱신

세션키와 보조키를 사용하여 그룹 키를 생성하는 절차는 다음과 같다(그림 1(b)).

1. BS는 그룹키를 갱신하기 위해 위의 단계(1)와 같이 각 노드들에 해당하는 키정보(KI_i)를 생성하고 자신의 메모리에 저장한다.
2. BS는 각 노드에게 KI_i와 T_s, T_{int}, N, $Auth_i$를 방송한다(그림 1(b)메세지(1)). 여기에서 KI_i'는 노드 i를 위한 키정보, N은 난수, $Auth_i$은 인증정보인 $MAC(SK_i, KI_i|T_s|N)$이다.
3. 메시지를 수신한 각 노드는 그룹키 정보에 대한 인증을 처리한다. 인증이 확인되면 자신의 세션키와 보조키를 사용하여 그룹키 GK'($=KI_i\oplus SK_i\oplus sK_i$)를 생성한다. 인증에 실패하면 BS에게 오류 메시지를 전송하고 수신한 메시지를 폐기한다.
4. BS는 각 노드에게 데이터를 보낼 때는 그룹키 GK'로 암호화하여 전송한다(그림 1(b)메세지(2)).

4.3 노드와 노드간의 서브 그룹키 생성



(그림 2) 2홉 이상 노드들 간의 임시키 설정

인접하고 있는 센서 노드와 노드간의 안전한 데이터 교환을 위해 그룹키를 사용할 수 있지만 데이터의 프라이버시를 보장하기 위해는 두 노드만이 공유하는 임시키($K_{A,B}$)를 사용할 필요가 있다. 임의의 노드가 자신의 1 홉 이웃에 있는 노드와 비화통신을 하기 위해서는 노드들이 임무현장에 배치된 후에 설정된 1 홉 임시키를 이용할 수 있다. 그러나 1 홉 이상의 거리에 있는 임의의 노드와 비화 통신을 하기 위해서는 다수의 중간 노드들을 포함해야 한다. 따라서 이 절에서는 초기에 설정된 1홉 임시키를 이용하여 1 홉 이상의 거리에 있는 노드들 간에 임시키를 설정하는 방법을 기술한다.

그림 2에서 노드 A는 노드 B와 1홉 임시키를 설정한 노드를 질의 메시지의 방송을 통해 검색한다(1). 이때 메시지는 질의 메시지 식별자(QRY: query identifier)와 키 설정 목표 노드(B)를 포함한다. 이 메시지를 수신한 노드 C와 D는 자신들이 노드 B와 이미 1 홉 임시키를 설정하였으므로, 응답메시지를 구성하여 노드 A에게 되돌린다(2). 이 응답메시지는 노드 A와 공유된 1 홉 임시키를 이용하여 암호화 한 뒤 노드 A에게 되돌려진다. 응답메시지를 수신하는 노드 A는 자신이 생성한 난수(N_A)와 수신한 난수들(N_C, N_D)을 XOR연산하여 노드 A와 B사이의 임시키를 생성한다. 이후 노드 A는 임의의 키 분할 방법을 이용하여 이 임시키를 응답노드의 수(즉, 2) 만큼 분할한다. 노드 A는 분할된 키들(K_{AB}^1, K_{AB}^2)을 각 응답노드(노드 C와 D)들에게 공유된 1 홉 임시키로 암호화하여 전송한다(3). 노드 C와 D는 수신한 부분키들을 다시 노드 B에게 암호화 하여 전송한다(4).

위에서 기술한 2홉 거리에 있는 노드들 간의 임시키 설정은 간단하게 3홉 이상의 거리에 있는 노드들 간의 임시키 설정으로 확장이 가능하다.

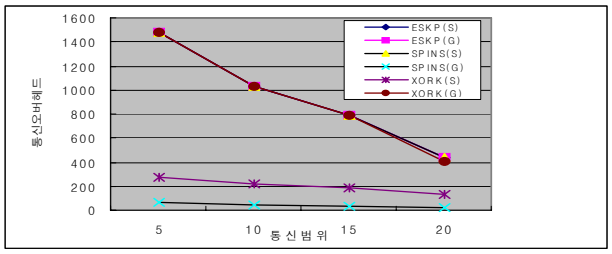
5. 성능분석

WSN 환경에서 시뮬레이션을 위하여 키 관리 프로토콜(ESKP, SPINS, XORK)들을 다음과 같은 가정 하에서 10초 동안 수행된다. 첫째, 하나의 BS 노드와 99개의 센서 노드들이 정해진 구역(100m×100m) 내에 불규칙하게 분포되어 있다. 둘째, 노드 간의 통신 범위의 경계선에 있는 특정 노드들을 클러스터 헤드로 지정하고 주위에 있는 노드들은 클러스터 헤드에게 데이터를 전송한다. 셋째, 편의상 각 프로토콜들은 세션키와 그룹키를 운용하고 인증을 수행한다. 마지막으로 메시지의 길이는 모두 같다. 정확한 시뮬레이션 결과를 얻기 위해 주어진 시간 동안에 노드간의 통신 범위를 5와 10, 15, 20으로 지정하였다. 그리고 매 시험 때마다 센서 노드들의 위치를 랜덤하게 재배치하였다.

5.1 네트워크 오버헤드

그림 3는 각 프로토콜(ESKP, SPINS, XORK)에

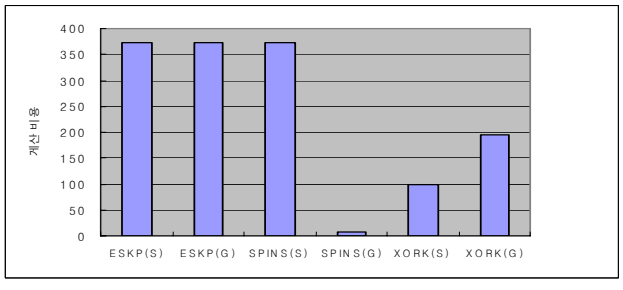
대해 키를 갱신하는 과정에서 통신 범위와 노드 간에 전송되는 전체 메시지 수의 관계를 나타낸 것이다. ESKP의 세션키와 그룹키, SPINS의 세션키, XORK의 그룹키를 갱신하는 과정에서 전송되는 메시지의 수는 거의 차이가 없지만 XORK의 세션키와 SPINS의 그룹키를 갱신하는 과정에서 전송되는 메시지의 수는 상당히 감소함을 알 수 있다.



(그림 3) 통신범위에 따른 네트워크 오버헤드

5.2 각 노드의 계산 오버헤드

그림 4는 각 프로토콜에서 노드들이 키 갱신 메시지를 처리하는 계산 비용을 나타낸다. 각 노드들은 수신한 메시지를 복호화하고 인증을 처리해야 하기 때문에 계산비용이 높게 나타났다. 하지만 XORK 프로토콜에서는 키를 복호화하기 위해 비트 위주 XOR 연산을 사용하기 때문에 기존의 복호화 알고리즘을 사용하는 것 보다 훨씬 비용이 적게 든다. 또한 제안된 프로토콜에서 세션키와 그룹키를 갱신하는데 있어서 계산 비용이 서로 다른 결과를 나타낸 것은 인증 처리 과정에서 전자는 단방향 해쉬함수를 사용하고, 후자는 MAC을 사용하기 때문으로 판단된다. 그리고 SPINS 프로토콜의 그룹키의 계산 비용이 아주 낮게 나타나는 것은 키를 암호화하지 않고 공개하는 메커니즘이기 때문이다. 그러나 지연된 키 공개의 방식은 간단하다는 장점을 가지고 있지만 일정한 시간이 경과되기 전에 키를 알 수 없다는 문제점을 가지고 있다.



(그림 4) 각 노드에서 계산 비용

6. 결론

본 논문에서는 데이터의 프라이버시를 보장할 뿐만 아니라 각 노드들의 전력 소비를 줄일 수 있는 키 관리 프로토콜을 제안하였다. 다시 말해서, 제안된 프로토콜은 대칭키 암호화 알고리즘을 사용함에 따라 발생할 수 있는 비밀키 노출 문제를 최소화하기 위해 정기적으로 키를 갱신하고 송신 메시지에 대한 인증을 수행하도록 하였다. 그리고 키를 갱신하는 과정에서 전송되는 메시지의 수를 가능한 한 줄이기 위해 송신자는 키 정보를 암호화하지 않고 공개된 하나의 랜덤 비트 패턴으로 방송하고, 세션키를 갱신하는 과정에서 처리 비용을 줄이기 위해 복호화 알고리즘을 사용하지 않고 간단하고 빠른 비트 위주 XOR 연산을 사용하였다.

제안된 키 관리 프로토콜과 다른 프로토콜을 비교해 본 결과 네트워크 내의 메시지 전송 횟수와 각 노드에서의 처리비용 측면에서 보다 우수함을 알 수 있었다. 그러나 메시지의 전송량과 계산비용을 줄일 수 있다는 측면에서 봤을 때 본 논문에서 제안한 프로토콜이 WSNs에 가장 적합한 구조라고 생각한다.

참고문헌

[1] H. Cam, S. Ozdemir, D. Muthuavinashiappan and P. Nair, “Energy-Efficient Security Protocol for Wireless Sensor networks,”in the Proceeding of the Hawaii International Conference System Sciences, 2004. pp. 173-186.

[2] D. W. Carman, P. S. Kruus and B. J. Matt, “Constrains and Approaches for Distributed Sensor Security,” NAI Labs Technical Report #00-010, Sep. 2000.

[3] Kamran Jamshaid and Loren Schwiebert, “SEKEN(Secure and Efficient Key Exchange for sensor networks),” IEEE 2004.

[4] L. Eschenauer and V. D. Gligor, “A Key-management Scheme for distributed Sensor Networks,” in proceedings of the 9th ACM Conference on Computer and Communications Security, Nov. 2002.

[5] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen and D. Culler, “SPINS:Security Protocols for Sensor Networks,” Wireless networks 8, 2002.

[6] G. Jolly, M. Kuscu and P. Kokate, “A Hierarchical Key Management Method for Low-Energy Wireless Sensor Networks,” UMBC Online Document, Nov. 2002.