키 체인을 이용한 무선 센서네트워크 키 분배 방식

김경준 박광규 송주석

연세대학교 컴퓨터과학과

Key Predistribution Schemes Using Key Chains for Wireless Sensor Networks

KyungJoon Kim, KwangKyu Park, JooSeok Song

Department of Computer Science, Yonsei University.

요 약

무선 센서 네트워크에서는 센서 노드의 제약으로 인해 공개키 알고리즘을 쓰기에는 적합하지 않다. 비밀키 방식을 사용할 경우 가장 문제가 되는 부분은 키 분배 문제이다. 한 가지 방법은 센서 노드들이 배치되기 전에 미리 센서 노드들에 키를 심어 놓음으로써 해결할 수 있다. 본 논문은 키 체인을 이용해 이러한 방식을 개선하여 더욱 향상된 보안성을 제공하는 프로토콜을 제시한다.

I. 서론[1)]

유비쿼터스 컴퓨팅 시대로의 도약을 앞두고, 무선 센서 네트워크(WSN)는 환경 모니터링에서부터 군사부분까지 다양한 분야에서의 응용이 기대되고 있으며, 현재 많은 연구가 진행 중에 있다. 무선 센서 네트워크는 self-organizing의 특성을 지닌 다수의 노드들로 구성된다. 넓은 지역에 임의로 배치되어 유지보수가 어렵고, 배터리로 구동되는 특성 때문에 전력 소비가 중요한 고려 대상이 된다. 제한된 프로세싱 성능으로 인해 RSA와 같은 공개키 기반의 암호화 시스템은 적용에 어려움이 있다. 무선통신을 사용하여 브로드캐스트되는 매체의 특성상 도청, 재밍(Jamming) 등의 공격에 취약하다. 다수의 노드 생산을 위해 저가에 생산하여야 하기 때문에 센서 노드에 대한 물리적인 공격(포획)에 취약하다는 특징을 가지고 있다. 특히 센서 노드의 배터리, 프로세싱능력 등의 한계로 공개키 알고리즘의 사용이 어려워 비밀키 방식이 사용되고 있는데 이 경우 키 분배 문제가 발생하게 된다. 한 가지 방법으로는 센서 노드들에 배치되기 전에 미리 노드에 키를 심어 놓음으로써 키 분배를 할 수 있다. 이를 pre-distribution pair-wise key방식이라 한다. 본 논문에서는 이미 제안된 pre-distribution pair-wise키 방식들을 살펴보고 키 체인을 이용한 향상된 보안성을 제공하는 프로토콜을 제시한다.

본문은 다음과 같이 구성되어 있다. 2장에서는 관련 연구들을 3장에서는 프로토콜을 제시하고 4장에서는 보안성 분석과 5장에서 비용분석을 하였다.

본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성·지원사업의 연구결과로 수행되었음

II. 관련 연구

1. Trivial Solution

기본적으로 생각할 수 있는 방식으로 센서 노드들이 뿌려지기 전에 미리 모든 노드에 master key를 넣어 놓는 것이다. 하지만 악의적인 공격자에 의하여 하나의 노드만이라도 획득이 되면 모든 네트워크에 관한 보안이 깨질 수 있다는 문제점이 있다. 다른 방식으로는 센서 노드마다 다른 노드와 사용하게 될 pair-wise키를 모두 넣어두는 방식이다. 네트워크에서 노드의 수를 N이라 하면 하나의 노드는 N-1개의 키를 가지고 있어야한다. 그래서 하나의 노드가 공격자에게 획득이 되더라도 공격자가 획득할 수 있는 정보는 해당 노드와 다른 노드들과의 링크 정보만으로 한정되어 피해를 최소화 할 수 있다. 하지만 노드마다 N-1개의 키를 가지고 있어야 되기 때문에 센서 네트워크의 크기가 커질수록 노드마다 요구되는 메모리가 늘어나 큰 네트워크에서는 적용하기 적합하지 않다.

2. Probabilistic Key Pre-distribution

[4] 논문에서는 센서 노드마다 다른 모든 센서 노드와의 pair-wise 키를 모두 가지고 있는 대신 key pool에서 어느 정도의 키만 뽑아서 가지고 있다가 노드가 배치된 후에 이웃에 있는 노드들과 키들을 비교하여 공통으로 가지고 있는 키를 pair-wise 키로 설정한다. 가지고 있는 키의 정보는 키 식별자를 이용하여 확인한다. 두 개의 노드가 하나 이상의 공통키를 가지고 있을 확률은 아래와 같다.

$$p = 1 - \frac{((P-k)!)^2}{((P-2k)!P!)}$$

여기서 P는 key pool의 크기이고 k는 노드 하나당 가지는 키의 수이다. 위 식을 통해서 두 노드가 공유키를 어느 정도 이상의 확률로 가지고 있기 위해서는 생각보다 적은 키만으로도 높은 확률을 달성할 수 있음을 보였다. 그리고 두 이웃 노드 간에 공유키를 발견하지 못한 경우는 이미 두 노드와 키가 설정된 다른 이웃노드들을 통하여 키를 설정한다.

3. Q-composite Key Pre-distribution

[4]논문을 보완한 논문이 [5]이다. 기존 방식은 이웃 노드 간에 여러 개의 공통키가 있는 경우 그 중 하나만을 pair-wise 키로 선택을 하였지만 이 논문은 이웃 노드 간 q개 이상의 공통키를 가질 경우에만 노드 간 pair-wise 키는 설정한다. pair-wise키는 두 노드의 공통 키들을 모두 Exclusive-OR 연산을 한 결과로 설정한다. 이 경우 공격자는 링크 간 교환되는 암호화된 정보를 해독하기 위해서는 q개의 공통키를 모두 알고 있어야 되기 때문에 이전 방식보다는 더 높은 보안성을 제공한다.

III. 제안하는 프로토콜

[5] 논문의 문제점은 다음과 같다. 노드 간 pair-wise 키가 성립되기 위해서는 공통으로 가지고 있어야 되는 키가 q개 이상 있어야 하기 때문에 공통키를 가지고 있을 확률을 높이기 위해서는 key pool의 크기를 줄이거나 노드마다 가지고 있어야 하는 key의 개수를 늘려야 한다. 하지만 이에 따라서 하나의 노드가 가지는 정보의 양이 증가하고 많은 수의 노드가 공격자에게 노출되었을 때는 오히려 [4]의 방법보다 더 많은 링크가 노출된다. 이러한 이유로 노드마다 같은 수의 키를 가질 때 [4]방식에 비해서 커버할 수 있는 네트워크의 크기는 줄어들게 된다. 위의 방식에서 발생하는 문제는 기본적으로 센서 노드의 메모리 제약 때문에 발생한다. 같은 메모리를 가지고 더 많은 키를 보유하는 방법이 있다면 이 문제를 어느 정도 해소할 수 있다. 첫 번째로 생각할 수 있는 방식은 key 하나의 크기를 줄이는 것이다. 이렇게 된다면 당연히 같은 메모리로 더 많은 키를 가질 수 있게 되지만 보안상으로는 brute-force attack등에 취약해 질 수 있다. 따라서 바람직한 방식은 아니다. 다른 방식은 여기에서 제안하는 키 체인을 이용하는 것이다. 이 방식은 key chain에서 임의의 레벨의 키를 선택하면 해당 키 체인에서 선택한 키의 레벨 이하의 모든 키들은 해시 함수를 이용하여 구할 수 있다는 것이다. 이를 식으로 표현하면 아래와 같다.

$$H_K(key_{c,l}) = key_{c,l-1} \quad (l > 1)$$

즉, $key_{c,l}$를 알고 있을 경우 $key_{c,0}$, $key_{c,1}$,…,$key_{c,l}$ 를 모두 구할 수 있음을 의미한다.

해시 함수의 특성상 $key_{c,l}$을 알고 있을 때 $key_{c,l+1}$을 구하는 것은 쉽지 않다. 따라서 자신이 가지고 있는 키 체인 레벨 이하의 키들만을 구할 수 있다. hash함수는 단순한 hash함수가 아닌 key를 사용하는 MAC(Message Authentication Code)을 사용할 것을 권한다. MAC의 경우 키가 추가로 사용되기 때문에 보안성을 높이고 새로운 키를 재분배 할 때도 유리하다.

키 설정 과정은 다음과 같다.

(1) 모든 노드들은 Key Pool에서 m개의 key chain을 선택하고 해당 키 체인에서 임의의 레벨의 키를 가진다.

(2) 노드들이 배치가 되면 서로 이웃노드들과 키 체인 정보를 교환한다.

(3) q개 이상의 공통 키체인을 가질 경우에만 pair-wise키를 설정하는데 pair-wise키는 공통 키 체인에서 두 노드가 가진 것 중 더 낮은 레벨의 키들의 exclusive-OR 연산을 한 결과로 키를 설정한다. 예를 들어, 노드 A가 키체인 1번의 레벨 3, 키체인 6번의 레벨 5, 키체인 9번의 레벨 1의 키를 가지고 노드 B가 키체인 6번의 레벨 4, 키체인 8번의 레벨 2, 키체인 9번의 레벨 5의 키를 가질 때, 형성되는 pair-wise키는 키체인 6번의 레벨 4와 키체인 9번의 레벨 1의 키의 exclusive-OR한 결과로 설정한다.

(4) 이 과정을 통해 pair-wise키가 설정되지 않은 노드들은 다른 방식과 마찬가지로 이웃 노드들의 도움으로 키를 생성할 수 있다.

이 방식에서도 두 노드 간에 pair-wise키가 생성될 확률은 q-composite random key 방식과 같다.

IV. 보안성 분석

키 체인의 최대 레벨이 L인 경우 노드가 레벨 l의 키를 가질 확률 함수를 $f(l)$이라 하자. 두 노드 사이에 pair-wise키를 생성할 경우 레벨 l의 키를 사용할 확률은 키 레벨이 같은 경우와 키 레벨이 다른 경우로 나누어 다음과 같이 구할 수 있다.

$$g(l) = f(l)^2 + 2f(l)\sum_{i=l+1}^{L} f(i)$$

하나의 키 체인이 노출 되었을 때 레벨 l의 키가 노출이 되지 않을 확률은 해당 키 체인에서 1~l-1 사이 레벨의 키를 가질 확률로 다음과 같다.

$$h(l) = \sum_{i=1}^{l-1} f(i)$$

하나의 노드가 공격자에게 노출이 되었을 때 그 중 특정 키 체인이 포함되어 있을 확률은 전체 key pool의 크기가 S이고 노드 하나당 m개의 키 체인을 가질 때 $\frac{m}{S}$가 된다. x개의 노드가 공격자에게 노출이 되었을 때 공격자에게 다른 링크들이 노출될 확률은 다음과 같이 구할 수 있다. x개의 노드가 노출되었을 때 하나의 키 체인이 노출된 노드 중 i번 포함이 되 확률은 $\binom{x}{i}\left(1-\frac{m}{S}\right)^{x-i}\left(\frac{m}{S}\right)^{i}$가 된다. 하나의 키 체인이 i번 노출이 되었을 때 레벨 l의 키가 노출이 되지 않을 확률은 $h(l)^i$가 된다. 그리고 레벨 l로 pair-wise키가 생성될 확률은 $g(l)$이 된다. 따라서 x개의 노드가 노출되었을 때 pair-wise키 생성에 사용된 하나의 키가 노출이 되지 않을 확률은 다음과 같다.

$$\sum_{i=0}^{x}\left\{\binom{x}{i}\left(1-\frac{m}{S}\right)^{x-i}\left(\frac{m}{S}\right)^{i}\sum_{l=1}^{L}h(l)^i g(l)\right\}$$

q개 이상의 공통 키 체인으로 pair-wise키를 생성한 경우 링크가 노출될 확률은 두 노드 사이에 가지고 있는 공통 키체인에서 모두 공통 키 체인 레벨 이상의 키를 가질 확률로 아래와 같다.

$$\sum_{j=q}^{m}\left[1-\sum_{i=0}^{x}\left\{\binom{x}{i}\left(1-\frac{m}{S}\right)^{x-i}\left(\frac{m}{S}\right)^{i}\sum_{l=1}^{L}h(l)^i g(l)\right\}\right]^j \frac{p(j)}{p}$$

여기서 $p(j)$는 두 노드 사이에 공통으로 j개의 공통 키 체인을 가지는 확률이고 $p = p(q)+p(q+1)+...+p(m)$ 이다.

V. 비용분석

계산양의 증가는 다음과 같다. 키 체인의 가장 높은 레벨의 키만 가지고 있는 경우 키 체인은 1~L 사이의 키 레벨을 가지고 키체인 하나당 평균 $\frac{L-1}{2}$만큼의 hash 연산을 하게 된다. 노드마다 m개의 키체인을 가질시 평균 $m\frac{L-1}{2}$의 해시 연산이 필요하다.

통신량의 증가는 다른 방식과 비교하였을 때 키(체인)의 식별자 번호뿐만이 아니라 레벨까지 알려주어야 하기 때문에 노드 하나당 가지고 있는 키 체인 수(m) × 레벨을 표현하기 위한 데이터비트 만큼의 비용이 발생한다.

메모리 부분의 비용은 hash함수를 추가로 사용하기 때문에 hash함수를 위한 메모리가 필요하다. 하지만 μTESLA등 hash함수를 사용하는 다른 프로토콜은 인증을 통해 키 폐지를 하는 등의 용도로 함께 사용될 확률이 매우 높다. 따라서 이러한 경우 hash함수를 사용하는 것은 추가 메모리 비용으로 보기 어렵다.

VI. 결론

본 논문에서는 키 체인을 이용한 향상된 보안성을 제공하는 프로토콜을 제시하였다. 본 논문에서 제시한 프로토콜은 앞에서 제시한 임의로 key를 분배하는 방식뿐만이 아닌 지역성 정보 등을 이용한 다른 key pre-distribution 방식에도 역시 적용할 수 있는 장점을 가진다. 문제점으로는 키 체인마다 레벨 조정을 전혀 하지 않았기 때문에 같은 q개의 키 체인을 가지고도 노드 간에 형성되는 링크의 보안성에 차이가 발생한다. 또한 키 체인의 높을 레벨의 키를 많이 가지고 있는 슈퍼 노드가 생성될 수도 있기 때문에 이러한 슈퍼노드들의 노출로 공격자는 훨씬 많은 정보를 얻을 수 있다는 문제점이 있다. 추후 이 부분을 보강하여 레벨을 일정하게 조정하는 프로토콜을 제안할 계획이다.

[참고문헌]

[1] E. Shi and A. Perrig, Designing Secure Sensor Networks, IEEE Wireless Communications, Dec. 2004.

[2] James Balasalle, Richard Han, Node Compromise in Sensor Networks, The Need for Secure System Carl Hartung

[3] C. Karlof, N. Sastry, D. Wagner, TinySec: A Link Layer Security Architecture for Wireless Sensor Networks, ACM SenSys 2004

[4] L. Eschenauer and V.D. Gligor, A Key-Management Scheme for Distributed Sensor Networks, ACM conference on Computer and Communications Security, Nov. 2002.

[5] H. Chan, A. Perrig, and D. Song, Random Key Predistribution Schemes for Sensor Networks, IEEE Symposium on Research in Security and Privacy, 2003.

[6] D. Liu, P. Ning, Multilevel μTESLA : Broadcast Authentication for Distributed Sensor Networks, ACM Transaction on Embedded Computing Systems, Vol. 3, No. 4, November 2004