DOI: 10.3745/KIPSTC.2009.16-C.2.183

무선 센서 네트워크를 위한 새로운 키 사전 분배 구조

김 태 연†

요 약

무선 센서 네트워크는 실세계에 광범위하게 배치되어 다양한 응용에 활용되고 있다. 센서 노드들 사이에 안전한 통신을 위한 필수 조건은 그들의 신뢰관계를 유지할 수 있는 세션키를 생성하는 것이다. 여기에서 고려되어야 할 문제는 어떻게 통신 노드들을 식별하고 키 동의 과정에서 상대방에게 키 정보의 노출을 최소화할 것인가이다. 현재 기존의 구조들에서는 몇 가지 취약점으로 인해 이러한 문제를 완전히 해결하지 못하고 있다. 따라서 본 논문에서는 다음과 같은 이점을 가진 새로운 키 사전 분배 프로토콜을 제안한다. 첫째, 노드간의 인증 서비스를 지원한다. 둘째, 공유하지 않은 키 스페이스의 식별자는 공개하지 않을 뿐만 아니라 공유하고 있는 식별자들의 공개를 최소화한다. 마지막으로 노드 공격에 대해 네트워크의 보안이 기존의 방식에 비해 강건하다. 그리고 성능과 보안 분석을 통해 제안된 구조가 무선 센서 네트워크에 적합함을 증명한다.

키워드 : 센서 네트워크, 비밀키, 협상키

A New Pairwise Key Pre-Distribution Scheme for Wireless Sensor Networks

Taeyeon Kim†

ABSTRACT

Wireless sensor networks will be broadly deployed in the real world and widely utilized for various applications. A prerequisite for secure communication among the sensor nodes is that the nodes should share a session key to bootstrap their trust relationship. The open problems are how to verify the identity of communicating nodes and how to minimize any information about the keys disclosed to the other side during key agreement. At any rate, any one of the existing schemes cannot perfectly solve these problems due to some drawbacks. Accordingly, we propose a new pre-distribution scheme with the following merits. First, it supports authentication services. Second, each node can only find some indices of key spaces that are shared with the other side, without revealing unshared key information. Lastly, it substantially improves resilience of network against node capture. Performance and security analyses have proven that our scheme is suitable for sensor networks in terms of performance and security aspects.

Keywords : Mobile Ad hoc Network, Data Consistency, Access Time, Time Constraint Applications

1. 서 론

무선랜 기술은 이동성과 빠른 전송 속도를 제공하면서 사용자들의 관심을 끌고 있다. 그러나 유무선 통합 망에 의해 제공되는 확장성 및 관리의 효율성을 제공받기 위해서는 반드시 무선 구간에서 취약한 보안기술을 해결해야 한다. 이에 따라 무선 랜이 더 영역을 넓혀 대규모의 망으로 사용되기 위해서는 데이터의 암호화, 인증, 무결성 등이 제공되는 보안 메커니즘이 필요하다. 특히, 무선 센서 네트워크(WSNs: Wireless sensor networks)는 자원의 취약성으로 인해 보안 수준이 그리 높지 못하다.

WSNs는 공간적으로 서로 다른 위치에서 온도나 습도, 압력, 움직임, 진동 등과 같은 환경적 조건들을 협력적으로 모니터하는 분산된 센서 장치들로 구성된다. 일반적으로 이러한 장치들은 무선 네트워크에서 노드의 역할을 하는 것으로 일정한 지역에 랜덤하게 배치된다. 노드들은 사전에 결정된 토폴로지로 구성되는 것이 아니라 배치된 후에 동적으로 다른 노드와 토폴로지를 구성할 수 있기 때문에 군사적 감시나 제어 통신, 재앙 지역 모니터링에 적합하다. 하지만 이러한 장치들은 근본적으로 낮은 비용과 낮은 전력, 다기능, 그 크기가 소형이며 근거리 통신을 지원하는 제약점을 가지고 있다.

분산된 센서 네트워크에서 임의의 두 노드 간에 안전한 링크의 설정을 보장하기 위해 Eschenauer[1]은 최초로 확률

† 종신회원 : 서남대학교 컴퓨터정보통신학과 조교수
논문접수 : 2008년 12월 4일
수 정 일 : 1차 2009년 1월 19일
심사완료 : 2009년 1월 22일

적 키 사전 분배 구조를 제안했다. 그리고 Chan[2]와 Du[3], Zhu[4] 등은 더 강한 보안성과 효율성을 지원할 수 있는 구조를 제안하였다. 하지만 이러한 구조들은 몇 가지 보안에 있어서 취약점을 가지고 있다. 첫째, 인접한 노드 간에 공유키의 유무를 확인하는 비밀 공유키 확인(shared-key discovery) 단계에서 공유키를 가지고 있는 노드가 거짓 데이터를 전송하거나 다른 노드로 가장하여 데이터를 전송하는 등의 문제가 있다. 둘째, 어느 한 노드의 비밀정보가 공격자에게 노출되는 경우 키 체인(key chain) 내의 대부분의 비밀키가 노출되는 취약점이 있다.

최근에 위의 두 번째 문제를 해결하기 위해 Chan[5]는 인접한 노드와 공유하지 않는 비밀키를 상대편 노드에게 공개하지 않는 상태에서 공유키를 확인하는 메커니즘을 제안하였다. 하지만 그들의 구조에서도 문제점들이 존재한다. 첫째, 공유키 확인 단계를 완료하기 전에 어느 악의 있는 노드는 인접한 노드가 관리하고 있는 키 체인내의 비밀키들 중에서 자신의 것들과 같은 것이 몇 개인지를 추측할 수 있으며, 인접 노드로부터 수신한 비밀 데이터의 취약점을 이용해서 상대편의 노드가 가지고 있는 키를 유도해 낼 수 있다. 그리고 이러한 구조들은 노드 인증을 지원하지 않는다.

Du[6]은 노드 인증 문제와 노드 공략(node capture)에 의한 공유키 노출 문제를 최소화하기 위해 Blom[7]의 구조를 기반으로 새로운 키 사전 분배 구조를 제안하였다. Eschenauer의 구조와 같이 BS(Base Station)은 모든 노드들에게 해당 위치에 배치되기 전에 다중 키 스페이스(multiple-key spaces)를 분배한다. 배치된 후에 임의의 두 노드 간에 안전한 통신을 위해서는 인접 노드와 하나 이상의 키 스페이스를 공유하고 있어야 한다. 이 구조에서 λ 개의 이하의 노드가 서로 공모(compromise)한다고 하더라도 전체 네트워크의 보안성은 보장된다. 그러나 이 구조 역시 두 가지 중요한 결함을 내포하고 있다. 첫째, 공격자가 행렬 Ac의 각 행의 값을 할당 받은 전체 노드 중에서 λ개 이상의 노드와 공모한다면 Ac의 행을 키 스페이스로 사용하는 모든 노드들과 통신할 수 있는 세션키를 생성할 수 있게 된다. 둘째, 인접한 노드들에게 통신 및 계산 부담을 가중시키기 위해 빈번한 거짓 메시지를 방송하여 무의미한 키 설정에 가담하도록 유도할 수 있다. 이러한 공격을 방지하기 위해서는 보다 엄격한 보안 정책이 필요하다.

따라서 본 논문에서는 센서 네트워크 환경에서 다음과 같은 장점은 가진 새로운 키 관리 구조를 제안한다. 첫째, 각 노드 간의 통신에 있어서 노드 인증을 보장한다. 둘째, 각 노드가 관리하고 있는 비밀정보의 노출을 최소화한다. 마지막으로 공격자에 의해 야기되는 통신과 계산 오버헤드를 최소화한다.

본 논문의 구성은 다음과 같은 순서로 구성된다. 1장의 서론에 이어서 2장에서는 관련 연구를 살펴본다. 3장에서는 제안된 세션키 생성 프로토콜 살펴보고, 4장에서는 성능 및 보안 분석에 관해 기술한다. 마지막으로 5장에서는 본 논문의 결론과 향후 연구방안에 대하여 기술한다.

2. 관련 연구

2.1 Chan의 구조

기존의 비밀키 사전 분배(key pre-distribution) 구조에서 비밀키들의 식별자를 교환하는 과정에서 키 정보가 노출되기 때문에 보안 문제가 야기된다. 하지만 Chan이 [5]에서 각 노드가 인접 노드들과의 세션키를 생성하는 과정에서 자신의 키 정보의 노출을 최소화하는 수정된 구조(MRS : Modified Rivest Structures)를 제안했다. 이 구조에서 사용하고 있는 비밀키 사전 분배 절차와 인접한 노드와 서로 공유하고 있는 키 확인 절차는 다음과 같다. 모든 노드는 해당 위치에 배치되기 전에 키 풀(key pool)에서 임의의 키 체인을 할당받는다. 그리고 해당 위치에 배치된 후에 각 노드는 안전한 채널을 설정을 하기 위해 인접한 노드와 서로 공유하고 있는 비밀키를 확인하는 절차를 수행한다. MRS 구조에서 세션키를 생성하기를 원하는 노드는 (식 1)과 같이 자신이 관리하고 있는 키 체인(예, $s_1, s_2, ..., s_m$)을 사용하여 비선형 다항식을 생성한다. 그리고 (식 2)와 같이 다항식의 각 계수를 자신의 비밀키로 암호화한 다음 이들 암호화된 계수들을 인접 노드들에게 방송한다. 인접 노드들은 수신한 데이터들을 사용하여 (식 2)와 같은 m차형 다항식을 생성한다. 그리고 다항식의 변수에 자신의 관리하고 있는 키 정보들을 각각 (식 2)에 대입하여 새로운 암호화된 데이터들을 계산한 다음 이 데이터들을 송신자에게 보낸다. 위 과정에서 수신한 데이터는 송신자만이 알고 있는 비밀키로 암호화되어 때문에 수신자는 복호화할 수 없다. 송신자 노드는 수신한 암호화된 값들을 복호화함으로써 인접한 노드와 서로 공유하고 있는 키의 존재 여부뿐만 아니라 무슨 키를 공유하고 있는지를 알 수 있게 된다. 다시 말해서 복호화된 결과가 '0'인 것은 수신자와 동일한 키 정보를 가지고 있다는 것을 의미하고, 그렇지 않은 것은 다른 키를 관리하고 있다는 것을 의미한다. 이러한 절차를 마치게 되면 송신자는 서로 공유하는 키 정보만을 공개한다. 송신자는 공유키 확인 과정에서 알게 된 비밀키를 사용하여 인접한 노드와 데이터를 교환하게 된다.

$$f_A(x) = (x - s_1)(x - s_2)...(x - s_m) = x^m + A_{m-1}x^{m-1} + ... + A_1x + A_0 \quad (1)$$

$$f_A^{'}(x) = x^m + E^{KA}(A_{m-1})x^{m-1} + ... + E^{KA}(A_0) \quad (2)$$

2.2 Du의 구조

Du[6]는 사전 키 분배 단계에서 Eschenauer이 제안한 방식[1]과 Blom의 구조[7]를 기반으로 해서 생성한 다중 키 스페이스를 해당 위치에 배치되기 전에 네트워크 내의 모든 노드(N)에게 분배하는 하는 프로토콜을 제안하였다. 다시 말해서 BS은 크기가 (λ+1)×N인 행렬 G와 크기가 (λ+1)×(λ+1)인 ω개의 대칭 행렬들(D1 ≠ D2 ≠ D3, ..., ≠ Dω)을 생성한 후에 행렬 G와 Dk(k=1, 2, ..., ω)을 사용하여 ω개의

새로운 행렬 Ai(키 행렬, = (Di • G)T)를 생성한다. 여기에서 행렬 G는 모든 노드에게 공개되는 정보로서 두 번째 행의 각 원소는 노드들의 ID를 나타낸다. 노드 x의 키 스페이스는 행렬 G의 x열에 대응되는 행렬 Ai의 행으로 정의한다. 여기에서 Ai를 생성하는데 사용되는 Dk의 k는 키 스페이스의 식별자라고 가정한다.

BS는 각 노드가 해당 위치에 배치되기 전에 ω개의 키 스페이스 중에서 m(<ω)개의 키 스페이스를 임의로 선택하여 분배한다. 각 노드가 해당 위치에 배치된 다음에, 임의의 두 노드가 하나의 세션키를 생성하고자 하는 경우에 각 노드는 자신의 ID와 키 스페이스의 식별자들을 서로 교환한다. 이 과정에서 같은 키 스페이스의 식별자를 관리하고 있는 노드 간에는 세션키를 생성할 수 있다. 예를 들어, 노드 x와 노드 y가 같은 키 스페이스의 식별자(예, c)를 가지고 있는 경우에 세션키를 생성 하는 절차는 다음과 같다. 세션키를 노드 x는 Kxy = Ac(y) × G(x)로, 노드 y는 Kxy = Ac(x) × G(y)로 계산하지만 두 계산 결과는 같다. 두 노드 간에 같은 키 스페이스의 식별자를 관리하고 있지 않는 경우에는 다른 인접 노드를 통해 세션키를 생성한다.

3. 제안된 세션키 생성 프로토콜

3.1 다중 키 스페이스(multiple key spaces) 생성

본 논문에서 네트워크의 보안성을 강화하기 위해서 다음과 같은 수정된 다중 키 스페이스 생성 절차를 사용한다. 첫 번째 단계에서 Du의 구조와 같이 BS는 크기가 (λ+1)×N인 행렬 G와 크기가 (λ+1)×(λ+1)인 ω개의 대칭 행렬(D1 ≠ D2 ≠ D3, ..., ≠ Dω)을 생성한 후에 행렬 G와 Dk(k=1, 2, ..., ω)을 사용하여 ω개의 키 행렬 Ai를 생성한다.

Du의 구조와는 달리 BS는 키 풀은 각 노드들이 세션키를 생성하는데 노드간의 최소의 연결성이 유지되고 상호간에 직접 연결되지 않는 경우에는 인접 노드를 통한 연결성이 보장되도록 구성한다. 다시 말해서 각 노드가 그들의 정보를 사용하여 그 인접한 노드들과 최소한 하나 이상의 세션키를 공유할 수 있는 확률로 키 풀을 구성한다. 또한 이 과정에서 공격자가 (λ+1) 이상의 노드가 공모하더라도 대칭 행렬 Dc를 계산해 낼 수 없도록 키 풀에 포함될 행렬 Ac의 행의 수를 제한한다. 따라서 키 풀의 크기를 키 스페이스의 개수로 정의한다면 최대 키 풀의 크기는 (행렬 A의 행 수)×(ω-1)이 된다. BS는 각 노드가 해당 위치에 배치되기 전에 키 풀로부터 랜덤하게 선택된 $m(2 \le m \le \omega)$ 개의 키 스페이스와 그 식별자, 키 스페이스에 해당하는 일방향 해쉬 함수를 각 노드들에게 분배한다. 해쉬 함수는 행렬 Ai에 따라 서로 다른 함수가 할당되기 때문에 m개의 함수가 사용된다. 해쉬 함수의 사용 목적은 공격자가 고의적으로 임의의 행렬 Ai의 식별자를 사용하여 정당한 사용자로 가장하는 공격을 방지하기 위함이다.

3.2 세션키 생성

3.2.1 협상(negotiation) 키

각 노드가 해당 위치에 배치된 후에 인접 노드와 세션키를 생성하기 위해서는 자신이 관리하고 있는 키 체인의 식별자와 같은 식별자를 관리하고 있는 인접 노드를 찾아야 한다. 이 과정에서 각 노드는 자신이 관리하고 있는 행렬 Ai의 식별자들을 인접 노드들과 안전한 채널을 통하여 교환해야 한다. 먼저, 송신자 노드(예, A)는 자신이 관리하고 있는 키 스페이스의 식별자들(예, $s_1, s_2, ..., s_m$)을 사용하여 (식 1)과 같이 m차 비선형 다항식을 생성한다. 그리고 생성된 다항식의 계수들을 (식 2)와 같이 자신의 비밀키(KA)로 각각 암호화하여 인접 노드들에게 방송한다.

A → all neighboring nodes :

$$E^{KA}(A_0), E^{KA}(A_1), ..., E^{KA}(A_{m-1})$$

수신 노드(예, B)는 수신한 암호화된 계수들을 사용하여 (식 2)를 생성한 다음 자신이 관리하고 있는 키 스페이스의 식별자들(예, t1,t2,...,tm)을 $f_B(x)$ 의 변수에 대입한 결과인 m개의 변형된 암호화된 값들을 노드 A에게 전송한다. 노드 A는 수신한 값들을 자신의 비밀키로 각각 복호화 함으로써 인접한 노드와 세션키를 생성할 수 있는지를 알 수 있다[5]. 하지만 최악의 경우에 노드 B는 노드 A에게 자신이 관리하고 있는 식별자들 중의 공유한 식별자 모두를 공개하는 결과를 초래한다. 또한 공격자가 정당한 사용자로 가장하여 임의의 값을 식별자로 지정하고 인접 노드들에게 방송하는 경우에 인접 노드는 자신의 비밀 정보의 일부분을 노출하는 문제를 야기할 수 있다. 이러한 문제를 해결하기 위해 본 논문에서는 키 스페이스의 식별자들을 변형하여 협상키를 사용한다. 각 식별자의 전반부와 후반부의 데이터를 일방향 해쉬 함수에 적용한 결과를 협상키로 사용하는 것이고, 식별자의 전반부의 데이터를 해쉬 함수에 적용한 결과와 식별자의 후반부를 협상키로 사용한 것을 나타낸 것이다. 예를 들어, 노드 A의 식별자가

$\{s_1(=a_{11} \| a_{12}), s_2(=a_{21} \| a_{22}), ..., s_m(=a_{m1} \| a_{m2})\}$ 이고, 노드 B의 식별자가

$\{t_1(=b_{11} \| b_{12}), t_2(=b_{21} \| b_{22}), ..., t_m(=b_{m1} \| b_{m2})\}$ 인 경우라고 하자. 여기에서 a_{i1} 와 b_{j1} 는 식별자의 비트 스트림에서 전반부를 의미하고, a_{i2} 와 b_{j2} 는 그 후반부를 의미한다. 노드 A의 협상 정보는

$\{s_{11}(=a_{11} \| h_{i'}(a_{11})), s_{21}(=a_{21} \| h_{j'}(a_{21})),$

$..., s_{m1}(=a_{m1} \| h_{k'}(a_{m1}))\}$ 와 $\{s_{12}(=h_{i'}(a_{12}) \| a_{12}),$ 이고, 노드 B의 협상 정보는 $\{t_{11}(=b_{11} \| h_{i''}(b_{11})),$ $t_{21}(=b_{21} \| h_{j''}(b_{21})), ..., t_{m1}(=b_{m1} \| h_{k''}(b_{m1}))\}$와 $\{t_{12}(=h_{i''}(b_{12}) \| b_{12}), t_{22}(=h_{j''}(b_{22}) \| b_{22}), ..., t_{m2}($

$= h_{k''}(b_{m2}) \| b_{m2})\}$ 이다. 여기에서 $1 \le i' \ne j' \ne k' \le \omega.$, $1 \le i'' \ne j'' \ne k'' \le \omega$ 이다.

3.2.2 세션키 생성

지금부터는 임의의 두 노드가 세션키를 생성하는 절차를 기술한다. 노드 A가 노드 B와 같은 스페이스를 사용하는 키 정보를 관리하고 있는 지를 확인하는 과정이라고 가정하자. 세션키를 생성하기를 원하는 노드 A는 그의 협상정보인 $s_{11}, s_{21}, ..., s_{m1}$ 을 생성한 다음 (식 1)에 대입한다. 그 다음에 노드 A는 비밀키 AK를 가지고 $f_A(x)$ 의 계수들을 암호화한 다음에 암호화된 결과인 $E^{AK}(A_0), E^{AK}(A_1), ..., E^{AK}(A_{m-1})$ 을 인접 노드들에게 방송한다. 역으로, 노드 A가 노드 B로부터 요청 메시지를 받았을 때는 협상 정보 $s_{12}, s_{22}, ..., s_{m2}$ 을 (식 4)에 대입하여 수정된 암호화된 데이터 리스트인 $r_0 f_B'(s_{12}), r_1 f_B'(s_{22}), ..., r_{l-1} f_B'(s_{m2})$ 을 생성한 다음 노드 B에게 전달한다. 여기에서 r_i 은 0이 아닌 서로 다른 수이다.

$$f_B(x) = (x - t_{12})(x - t_{22})...(x - t_{m2}) = x^m + B_{m-1}x^{m-1} + ... + B_0 \quad (3)$$

$$f_B'(x) = x^m + E^{KB}(B_{m-1})x^{m-1} + ... + E^{KB}(B_0) \quad (4)$$

노드 B가 인접 노드들에게 메시지를 방송할 때는 자신의 협상 정보를 생성한 다음 (식 3)에 대입을 한다. 그리고 노드 B는 $f_B(x)$ 의 계수를 자신의 비밀키 BK로 암호화한 결과인 $E^{BK}(B_0), E^{BK}(B_1), ..., E^{BK}(B_{m-1})$ 을 인접 노드들에게 방송한다. 역으로 다른 노드로(예, A)부터 요청 메시지를 받았을 경우에는 식 2에 응답 협상 정보를 대입하여 얻어낸 수정된 암호화 데이터 리스트인 $r_0' f_A'(t_{12}), r_1' f_A'(t_{22}), ..., r_{m-1}' f_A'(t_{m2})$ 을 해당 노드에게 전송한다. 여기에서 r_i' 은 0이 아닌 서로 다른 수이다. 자세한 수행절차는 다음과 같다.

노드 A는 인접 노드들에게 $f_A(x)$ 의 암호화된 계수들을 방송한다. 메시지를 수신한 노드 B는 비밀키(KA)를 모르기 때문에 수신한 계수들을 복호화할 수 없다. 그리고 노드 B는 수정된 암호화된 계수인 $M' = \{r_1' f_A'(t_{11}), r_2' f_A'(t_{21}), ..., r_m' f_A'(t_{m1})\}$ 와 $f_B(x)$ 에 자신이 생성한 협상 정보를 암호화한 계수들을 노드A에게 보낸다. 노드 B로부터 M' 를 수신한 노드 A는 M' 을 복호화($D^{KA}(r_i' f_A'(t_{i1}))$)하고, $D^{KA}(r_i' f_A'(t_{i1}))$ 이 0이면 1로 그렇지 않으면 0으로 표시한 m-비트-맵을 생성한다. 여기에서 i-번째 비트가 1인 경우는 노드 B가 (식 2)에 대입한 t_{i1} 이 자신의 협상키들 중의 하나의 전반부와 일치한다는 것을 의미한다. $D^{KA}(r_i' f_A'(t_{i1}))$ 가 0 이 아닐 경우, 노드 A는 r_i' 을 모르기 때문에 t_{i1} 가 무엇인지를 정확히 알지 못한다. 그리고 보안성을 강화하기 위해서 m-비트-맵 내의 1이 하나 이상인 경우(예, W)에 랜덤하게 비트 값을 w (=$\left\lceil \frac{W}{2} \right\rceil < m$) 개로 변경한 m-비트-맵과 $M = \{r_1 f_B'(s_{12}), r_2 f_B'(s_{22}), ..., r_m f_B'(s_{m2})\}$ 을 노드 B에게 전송한다.

다시 노드 A로부터 메시지를 받은 노드 B는 m-비트-맵을 통해 식별자들의 전반부의 값이 같은 것이 무엇인지를 알 수 있다. 그리고 식별자의 전반부가 같은 $D^{KB}(r_i f_B'(s_{i2}))$ 을 복호화한 결과가 0인 것들을 확인한다. 0인 경우가 하나 이상인 경우는 앞에서 설명한 방식으로 그 개수를 조정하여 m''-비트-맵을 노드 A에게 전송한다. 최종적으로 각 노드는 m-비트-맵과 m'-비트-맵을 사용하여 세션키 ($SK = h_1(K_1 \| K_2 \| ... \| K_j)$)를 생성한다.

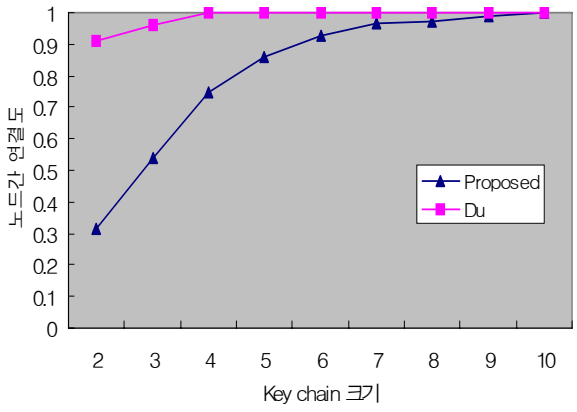
4. 성능 및 보안성 분석

이 장에서는 Du[6]에 의해서 제안된 구조(Du)와 본 논문에서 제안된 구조에 있어서 노드의 연결성과 보안성에 관하여 분석한다. 제안된 구조의 이용성과 보안성을 평가하기 위해 두 가지 측면에서 비교 분석한다. 즉, 비밀키 확인 단계 동안에 두 인접한 노드는 적어도 하나의 키 스페이스를 공유할 확률과 키 스페이스들의 식별자가 노출될 확률에 대해 분석한다. 본 시뮬레이션의 네트워크 모델은 다음과 같이 가정한다. (a) 150개의 노드가 지역 100×100㎡에 랜덤하게 배치된다. (b) 각 스페이스를 위한 보안 매개 변수인 λ의 값은 15이며, BS가 생성하는 키 스페이스의 수는 30이다. (c) 키 체인의 크기(m)는 3부터 10까지이다. (d) 전송 범위는 모든 노드에 대해 일정하다고 가정한다.

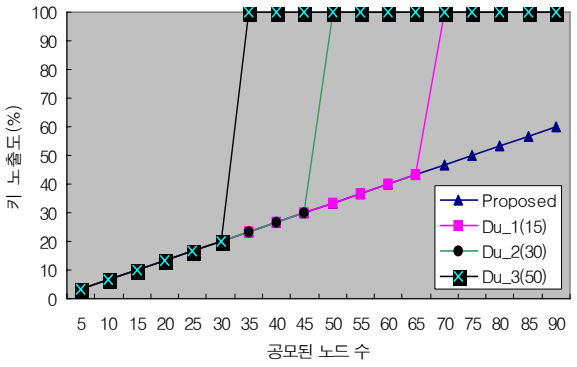
4.1 성능 분석

4.1.1 키 스페이스 공유 확률

본 논문에서는 비교적 저렴한 대칭키 암호 알고리즘을 사용하기 때문에 계산 복잡성은 고려하지 않는다. 로컬 노드의 연결성은 키 풀에 포함될 행렬 Ac의 행의 수와 키 체인의 크기에 의해 좌우된다. (그림 1)은 ω가 30이고, 키 체인의 수가 3에서 10으로 변할 때 노드의 연결성을 나타낸 것이다. (그림 1)에서 보이는 것처럼 키 체인의 수가 증가할수록 임의의 두 노드간의 연결 확률은 역시 증가한다. Du의 구조에서는 키 체인의 크기에 노드간의 연결성에 큰 영향을 미치지 않지만 제안된 구조에서는 키 체인의 크기가 작은 경우에는 노드간의 연결성이 낮음을 알 수 있다. 따라서 노



(그림 1) Key chain의 크기에 따른 노드간의 연결도



(그림 2) 노드의 공모에 따른 키 노출

드 연결성을 0.5이상이 되게 하기 위해서는 키 체인의 크기를 3이상으로 하는 것이 타당할 것으로 판단된다.

4.1.2 통신 오버헤드

통신 오버헤드는 주로 송수신되는 메시지의 크기나 수에 의해 결정된다. 일반적으로 확률적 키 사전 분배 구조들은 인접 노드들과 공유 정보를 찾기 위해서 각 노드는 인접 노드들과 그의 키 정보를 교환한다. 이러한 절차를 위해 Eschenauer등의 구조에서는 암호화된 리스트(($\acute{a}$, $E^{Kv}(\acute{a})$), v = 1, …, m)를 인접 노드들에게 전달하지만 Du 등의 구조에서 송신 노드는 키 스페이스의 식별자들을 평문으로 전송한다[1,6]. 그러나 본 논문에서 제안한 구조에서는 협상키를 사용하기 때문에 키의 식별자나 키 스페이스의 식별자를 평문으로 사용하는 기존의 구조보다 계산 비용에서나 통신비용에 있어서 다소 오버헤드가 증가한다. 그러나 기존의 구조에서 공격자가 인접 노드가 관리하고 있는 키 정보를 알아내기 위한 공격이나 서비스 공격을 하기 위해 임의의 정보를 생성하여 인접 노드들에게 빈번하게 방송을 하는 경우에 그 계산 비용과 통신비용은 상당히 증가하게 된다.

4.2 보안 분석

공격자가 동일한 행렬 A로부터 생성된 키 스페이스를 사용하는 λ개 이상의 노드와 서로 공모하는 경우에 나머지 노드들에 치명적인 보안 문제를 야기하기 때문에 각 노드는 자신의 키 정보를 안전하게 관리해야할 뿐만 아니라 전송되는 키 정보가 공격자에게 노출되는 것을 최소화하는 것이 매우 중요하다. 이와 관련하여 이 절에서는 제안된 프로토콜을 사용하여 노드간의 키 정보를 전송하는 과정에서 발생할 수 있는 키 노출 정도를 분석한다. 본 논문에서 제안한 구조는 Du의 구조와는 달리 공유하지 않은 스페이스의 식별자를 공개하지 않을 뿐만 아니라 공유된 식별자의 수가 2개 이상인 경우에 그 수를 제한한다.

전술한 바와 같이 기존의 구조에서는 키 스페이스의 확인 단계가 종료된 후에 각 노드는 인접한 노드들의 키 스페이스(행렬 Ac의 행)들은 알 수 없지만 그들이 관리하고 있는 모든 식별자들을 알 수 있게 된다. 이러한 경우에 어느 한 노드의 비밀정보가 공격자에게 노출되는 경우, 같은 식별자를 관리하는 노드들은 나중에 쉽게 공격을 받을 수 있다. 최악의 경우, 즉 공격자가 (λ+1) 이상의 노드들과 공모했을 경우에 통신하고자 하는 모든 노드들과 세션키를 생성할 수 있게 된다. 하지만 제안된 구조는 협상키를 사용함으로써 이러한 문제를 최소화할 수 있다. (그림 2)는 행렬 Ac의 크기가 100이고 보안 매개변수(λ)가 15, 키 풀의 크기(ω)가 30인 경우에 공유된 키 스페이스의 식별자의 노출 정보를 나타낸 것이다. 여기에서 Du_1(15)와 Du_2(30), Du_3(50)은 전체 생성 가능한 키 스페이스 중(100)에서 각각 15와 30, 50개만을 임의의 노드에게 분배한다는 의미이다. (그림 2)에서 보는 바와 같이 전체 노드 중에서 30개 이하의 노드가 공격을 받았을 경우에는 기존의 방법과 그 차이를 보이지 않는다. 즉, 공격받은 노드이외의 노드에게는 키 노출 문제를 야기하기 않는다. 하지만 Du_3(50)의 경우에는 30개 이상의 노드가 공격자로부터 공모되었을 경우에는 전체 노드에게 영향을 미친다는 것을 나타낸다. 또한 Du_2(30)와 Du_3(15)의 경우에는 공모되는 노드의 수가 각각 45와 65이하에서는 안전하지만 그 이상이 되었을 경우에는 다른 노드들의 키 스페이스를 쉽게 생성할 수 있는 행렬(Dc)을 생성할 수 있음을 나타낸다. 반면에 제안된 구조는 공격받은 노드 수에 비례한다는 것을 나타내고 있다. 결과적으로 제안된 구조에서는 전체 노드가 공모하지 않는 한 대칭 행렬(Dc)을 생성할 수 있는 가능성이 매우 낮음을 알 수 있다.

5. 결 론

본 논문은 센서 네트워크에서 안전한 비밀키 사전 분배 구조를 제안한다. 보안성을 강화하기 위해 본 논문에서는 수정된 다중 스페이스 생성 메커니즘과 기존의 구조들에서 사용하고 있는 키의 식별자나 행렬의 식별자를 사용하는 대신 협상 정보를 사용하여 한 쪽 노드뿐만 아니라 두 노드들이 관리하고 있는 키 정보들의 노출을 최소화하는 구조를 기술하였다. 또한 노드의 인증을 보장하고 비밀키 확인 과정에서 다중 랜덤 변수를 사용하여 암호화 알고리즘의 취약

점을 보완하였다. 따라서 제안된 구조는 두 노드가 안전한 채널을 통해 세션키의 설정을 보장하고 공격자에 의한 노드 공격에 대한 노드의 강건성을 보장한다. 시뮬레이션 결과는 보안성의 측면에서 제안된 구조가 기존의 Du의 구조에 비해 더 우수함을 보여주고 있다. 향후 필요한 연구 과제는 실제 무선 센서 네트워크 환경에서 제안된 기술을 구현하는 것이다.

참 고 문 헌

[1] L. Eschenauer, and V. D. Gligor, “A Key-management Scheme for Distributed Sensor Networks,” Proc. of the 9th ACM Conference on Computer and Communications Security, pp.41-47, 2002.

[2] H. Chan, A. Perrig, and D. Song, “Random Key Pre-distribution Schemes for Sensor Networks,” IEEE Symposium on Research in Security and Privacy, May, 11-14, pp.197-213, 2003.

[3] W. Du, J. Deng, Y. S. Han and P. K. Varshney, “A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge,” IEEE INFOCOM (2004), pp.586-597, 2004.

[4] S. Zhu, S. Setia, S. Jajodia, “Establishing Pairwise Keys for Secure Communication in Ad Hoc Networks: A probabilistic Approach,” Proceedings of the 11th IEEE International Conference on Network Protocols (ICNP’03), pp.1-10, 2003.

[5] A. C-F. Chan, “Distributed Symmetric Key Management for Mobile Ad hoc Networks,” IEEE INFOCOM pp.2414-2424, 2004.

[6] W. Du, J. Deng, Y. S. Han and P. K. Varshney, “A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks,” Proc. of ACM Conference on Computer and Communications Security (CCS’03) pp.42-51, 2003.

[7] R. Blom, “An Optimal Class of Symmetric Key Generation System,” Advances in Cryptology: Proceeding of EUROCRYPT 84, lecture Notes in Computer Science, Springer-Verlag, pp.335-338, 1985.

김 태 연

e-mail : tykim@seonam.ac.kr

1988년 전남대학교 계산통계학과(이학석사)

1996년 전남대학교 전산통계학과(이학박사)

1996년~1998년 서남대학교 전산정보학과 전임강사

1999년~현 재 서남대학교 컴퓨터정보통신학과 조교수

관심분야 : 정보보안, 무선센서네트워크, 네트워크 관리 등