무선센서네트워크에서 다항식 비밀분산을 이용한 공개키 인증방식에 관한 연구

김일도* · 김동천*

A study on Public Key Authentication using Polynomial Secret Sharing in WSN

Il-do Kim* · Dong-cheon Kim*

요 약

센서네트워크의 인증과 관련된 초기의 연구에서는 센서노드의 자원제약적인 특징을 고려하여 대칭키 기반의 인증 방식이 주로 제안되었으나, 최근에는 암호알고리즘의 성능이 개선되고 센서노드의 제조기술이 발달하여 Merkle 트리 방식 등 공개키 기반의 인증 방식도 제안되고 있다. 따라서 본 연구에서는 센서네트워크에 효과적으로 적용될 수 있는 새로운 개념의 다항식 비밀분산을 이용한 공개키 인증방식을 제안하며, hash 함수를 이용한 악의적 노드탐지 기법도 제안한다. 제안된 인증방식은 Shamir의 임계치 기법에 변형된 분산정보의 일종인 지수(exponential) 분산정보 개념을 적용하여 동시에 주변 노드들을 인증하면서 센서노드의 자원을 최소로 사용하고 네트워크의 확장성을 제공한다.

ABSTRACT

Earlier researches on Sensor Networks preferred symmetric key-based authentication schemes in consideration of limitations in network resources. However, recent advancements in cryptographic algorithms and sensor-node manufacturing techniques have opened suggestion to public key-based solutions such as Merkle tree-based schemes. This paper proposes a new concept of public key-based authentication using Polynomial Secret Sharing that can be effectively applied to sensor networks and a detection of malicious node using the hash function. This scheme is based on exponential distributed data concept, a derivative from Shamir's (t,n) threshold scheme, in which the authentication of neighbouring nodes are done simultaneously while minimising resources of sensor nodes and providing network scalability.

키워드

센서네트워크, 공개키 기반 인증, (t,n) 임계치 기법

Key word

sensor network, public key-based authentication, Shamir's (t,n) threshold scheme

* 해군사관학교

접수일자 : 2009. 05. 21
심사완료일자 : 2009. 07. 07

Ⅰ. 서 론

무선센서네트워크(WSN: Wireless Sensor Network, 이하 WSN)에서 인증을 위한 초기의 연구들은 비교적 연산속도가 빠른 대칭키 기반의 인증에 중점을 두고 진행되었다. 최근에는 센서 노드 제조기술의 발달로 공개키 기반의 암호 시스템이 적용 가능해짐으로써 Merkle 기반의 인증과 같은 공개키 기반의 노드 인증 기법들이 제안되고 있다.

그러나 기존 인증 기법들은 대부분 노드 상호 간 일대일 인증을 요구하고 있기 때문에 대량의 센서 노드들 간의 인증을 위해서는 상당히 많은 시간이 소요될 뿐만 아니라, 인증과정에서 발생하는 통신오버로드가 네트워크의 크기에 따라 증가한다는 문제점을 안고 있다.

공개키 기반의 인증 기법들 역시 유사한 문제점을 안고 있다. WSN은 기반구조가 없이 구성되는 네트워크임에도 불구하고 중앙 집중적인 인증기능을 수행하는 인증기관의 구현을 제안한다거나 혹은 필드에 새로운 센서 노드들이 추가 배치될 경우 확장성을 제대로 지원하지 못하는 문제점이 있다. 비교적 다른 인증 기법들에 비해 효율성을 인정받은 Merkle 트리 기반의 인증도 네트워크의 크기에 비례해서 메모리 사용량과 통신 오버로드가 증가하며, 새로운 노드의 추가 시 인증 트리를 재구성해야 하는 등의 문제점이 있어 자원 제약적 특성을 가지는 WSN에는 적합하지 않은 인증 기법이라 할 수 있다.

본 연구에서는 WSN의 차별화된 네트워크 특성들과 센서 노드의 자원 제약적 특성을 최대한 반영하면서 기존 인증 기법들의 문제점을 보완할 수 있는 공개키 기반 인증방식을 제안한다. 제안된 인증 기법은 적대적 운용환경 하에서 인증을 방해하는 공격 행위를 했을 경우, 해당 공격 노드를 조기에 발견하여 네트워크에서 배제시키는 메커니즘을 포함한다.

Ⅱ. 관련연구

2.1 WSN에서 공개키 기반의 인증 기법

WSN에서는 유선망에서와 같이 인증서를 발급, 분배, 관리하는 인증기관의 문제를 해결하기 위해 중앙 분배 방식을 이용할 수 없는 환경이다. 그래서 기반체계가 없는 WSN에서는 일대일 키 관리 문제를 공개 키 기반 암호시스템의 분산기법을 이용하여 해결하고자 하는 연구들이 주로 제안되고 있다. 분산방식은 다시 완전분산기법과 분산서브그룹기법으로 구분된다. 하지만 실제적으로 완전분산기법은 수많은 노드로 구성된 WSN에서는 적용하기에 문제점이 있어 분산서브그룹기법을 위주로 주요연구가 진행되고 있다.

인증기관을 여러 개의 다른 노드들에게로 분산하는 기법[1]은 (k,n)임계치 기법을 사용한다. 인증기관의 부분 비밀키를 n개의 노드가 나누어 가지고, 각 노드는 자신이 가지고 있는 비밀키를 이용하여 서명한 부분 인증서를 만들 수 있게 된다. 그리고 이중에서 k개 이상의 노드들이 모인다면, 완전하고 적법한 인증서를 만드는 인증기관의 기능을 수행할 수 있게 된다.

이 기법은 중앙 분배 방식의 인증기관의 기능을 분산시켰으므로 WSN 환경에 적합하지만, 이 기법의 변수인 k와 n의 값이 인증기관에서는 기능의 가용성, 전체 시스템의 안전성, 그리고 비용에 큰 영향을 미치므로, 여러 가지 상황들을 고려해서 변수의 값들을 선택하는 것이 중요하다. 그리고 부분 비밀키를 가지고 있는 노드들이 서로 원활하게 통신할 수 있도록 노드들의 위치도 신중하게 고려되어야 한다.

인증기관이 없이 네트워크에 참여한 노드들이 스스로 인증서의 문제를 해결하는 방법들도 연구되었다. 대표적인 방법으로는 유선망의 PGP(Pretty Good Privacy)와 같이 각 노드들이 스스로 인증서를 발급하고 저장하여 자신이 한 노드를 인증하면 그 노드가 인증하는 다른 노드까지 인증서 사슬을 통해 서로를 확인할 수 있는 방법[2]과 주변의 노드들과만 통신할 수 있는 채널로 미리 주고받은 데이터를 이용하여 서로의 공개키에 대한 해쉬 값을 주고받는 과정을 통해 인증하는 방법[3]이 있다.

첫 번째 방법은 발급한 인증서의 수가 많지 않은 초기 단계에서 인증이 제대로 이루어지지 않고, 많은 인증서를 갖고 있기 때문에 메모리 한계가 있다. 두 번째 방법은 주변의 노드들과의 완전한 통신채널 필요성으로 국지적인 환경에서만 적용이 가능하다는 문제가 있다.

2.2 Merkle 트리를 이용한 공개키 인증

Du 등은 Merkle의 인증 트리 기법을 이용하여 해쉬 함수만으로 효율적인 공개키 인증이 가능한 기법을 제

안[4]하였으며, Merkle 트리는 할당(assignment) Φ를 통해서 내부 노드 및 루트노드로 구성된 완전이진트리 구조이다. 그림 1에서 보여주는 바와 같이 Merkle 트리를 구축하기 위해 N개의 노드로 구성되는 WSN에서는 N개의 잎 노드(leaf node)로 구성된다. 잎 노드와 그 외의 노드들이 가지게 되는 Φ값은 다음 식에 의해 정의된다.

$$\Phi(L_i) = h(id_i, pk_i), \text{ for } i = 1, \cdots, N$$

$\Phi(L_i)$는 i번째 노드의 식별자와 공개키를 해쉬 함수로 매핑한 값이고, 후에 계산될 $\Phi(R')$와 $\Phi(R)$값이 동일할 경우, 사용자의 공개키를 인증하는 방식이다. 여기서, V는 전체 Merkle 트리에서 내부 노드를 가리킨다. 부모 노드들은 자식 노드들의 Φ값의 접합을 다시 해쉬하여 자신의 Φ값을 구한다. 루트 노드도 다른 부모 노드와 동일하게 자신의 Φ값을 구한다. Merkle 트리의 생성이 완료되면 모든 노드는 Merkle 트리의 루트인 $\Phi(R)$과 각 노드(잎 노드)를 기준으로 루트까지의 경로 λ에 포함되는 노드들의 형제 Φ값들을 부가인자로 저장한다. 이에 따른 각 노드의 메모리 사용량은 $(\log_2 N)+1$이 된다.

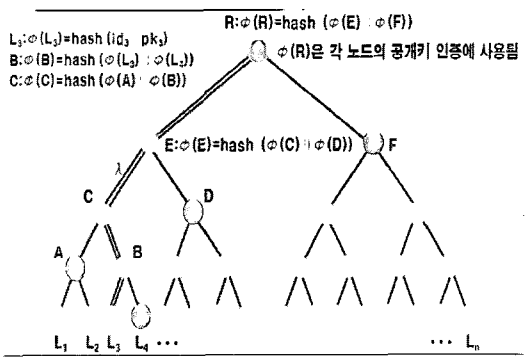

그림 1. Merkle 트리를 이용한 공개키 인증
Fig. 1 Public key authentication using Merkle Tree

2.3 비밀분산기법

비밀분산(Polynomial Secret Sharing)이란 특정 그룹의 참가자들에게 비밀을 분산시키는 것으로서 각 참가자는 분산정보가 할당된다. 이 비밀은 분산정보가 모아져야만 복구될 수 있으며 각각의 분산정보만으로는 비밀에 대해서 아무 것도 알아 낼 수가 없다. 이러한 비밀분산은 비밀키의 안전한 저장, 공개키를 이용한 암호화된 백업시스템 구축, 여러 통신경로로 비밀 메시지 전송, 권한 분산 등의 용도로 사용되고 있다[5].

Ⅲ. 비밀분산을 이용한 노드 인증

3.1 (t,n) 임계치 기법과 지수비밀 이용 노드 인증

Shamir의 (t,n) 임계치 기법[6]에서 사용되는 Lagrange 보간 다항식 기법을 이용하여 인증에 필요한 분산정보를 생성하고 인증하는 기본 방식을 설명하고 ElGamal 암호시스템[7]을 이용한 변형된 지수분산정보의 생성방법을 제안하며, 생성된 지수분산정보를 이용한 새로운 개념의 공개키 및 노드 인증과정에 대하여 설명한다.

3.1.1 Lagrange 다항식을 이용한 분산정보 생성 및 인증

특정 그룹의 n명의 참가자들에게 비밀에 대한 분산정보를 배포하는 배포자는 아래와 같은 $t-1$차 Lagrange 다항식을 임의로 생성한다. $a,b,c,\cdots$는 임의의 상수, K는 비밀, p는 분산정보보다 큰 소수이다.

$$F(x) \equiv ax^{t-1} + bx^{t-2} + cx^{t-3} + \cdots + K \bmod p$$

배포자는 타원곡선 암호방식 등의 공개키 암호시스템을 이용하여 i번째 참가자의 공개키(pk_i)와 개인키(sk_i)의 공개키 쌍 (pk_i, sk_i)을 생성하고 각 참가자에게 한 쌍씩 할당한다. 각 참가자의 공개키(pk_i)를 Lagrange 다항식 $F(x)$의 x값에 대입하여 다음 식과 같이 각 참가자의 분산정보 K_i를 계산한다.

$$K_i = F(pk_i)$$

각 참가자들은 동일한 비밀 K, 각각의 공개키 쌍 (pk_i, sk_i)과 분산정보 K_i를 저장하고 신원 미상의 상대를 인증할 필요가 있을 경우 상대방의 공개키 pk_i, 분산정보 K_i 정보를 요청한다.

제안하는 공개키 인증방식은 한 번에 특정 상대방 1명만을 인증하지는 못하고 $t-1$명의 상대방을 동시에 인증하는 방식을 취하고 있다. 따라서 인증작업을 수행하기 위해서는 최소한 t개의 (pk_i, K_i) 정보가 필요하다.

참가자 본인의 (pk_i, K_i)와 다른 참가자 $t-1$명의 (pk_i, K_i) 정보를 가지고 있으면 다음과 같은 식을 이용하여 Lagrange 다항식 $F'(x)$를 생성할 수 있다.

$$F'(x) = \sum_{s=1}^{t} K_s \prod_{j=1, j\neq s}^{t} \frac{(x - pk_j)}{(pk_s - pk_j)} (\mathrm{mod} p)$$

$F'(x)$가 배포자가 그룹의 참가자들에게 분산정보를 배포할 때 사용하던 Lagrange 다항식 $F(x)$와 동일한지를 알아보기 위하여 $F'(x)$의 상수항 K'가 다항식 $F(x)$의 비밀 K와 동일한 값인지 확인하면 된다. 만약에 $K' = K$이라면 이 다항식 $F'(x)$를 생성하기 위하여 사용된 $t-1$명의 상대방이 자신과 동일한 다항식 $F(x)$를 사용하여 (pk_i, K_i) 정보를 생성했다고 확인할 수 있는 것이다. 이와 같이 $t-1$명의 정보가 정상적인 공개키와 분산정보임을 확인함으로써 $t-1$명의 공개키 (pk_i)를 동시에 인증하고 해당 노드들과의 신뢰관계를 형성할 수 있게 된다.

그러나 인증과정에서 각 참가자의 인증정보가 평문으로 브로드캐스팅되는 경우에는 심각한 위험이 발생할 수 있다. p, t값을 미리 알고 있는 공격자가 평문으로 전송되는 t개의 인증정보를 탈취하게 되면 이를 이용하여 Lagrange 다항식 $F(x)$를 쉽게 알아낼 수 있게 된다. 그리고 인증정보를 비밀리에 전달하였다고 하더라도 t명이 공모하거나 공격자에 의해 포획되었을 경우에는 $F(x)$를 쉽게 알아 낼 수 있기 때문에 인증체계가 무너질 수 있다는 문제점이 있다.

3.1.2 변형된 분산정보와 ElGamal 암호시스템

가. 변형된 분산정보의 개념

Shamir의 (t,n) 임계치 기법[8]에서 $t-1$차 다항식 F에서 $F(0)$는 비밀 K이다. 각 특정 그룹의 참가자 i의 분산정보는 $K_i = F(i)$로 주어진다. n명 중 t명의 참가자로 이루어진 부분집합을 B라고 하면 π_B: $B \rightarrow 1,2,\cdots,n, |B| = t$ 이다. t개의 분산정보$(K_{\pi_B(1)}, K_{\pi_B(2)}, \cdots, K_{\pi_B(t)})$로 이루어진 임의의 부분집합 B를 이용하여 다항식 F를 다음과 같이 복원할 수 있다. 여기에서 계산은 $GF(p)$상에서 이루어진다.

$$F(x) = \sum_{s=1}^{t} K_{\pi_B(s)} \prod_{j=1, j\neq s}^{t} \frac{(x - x_{\pi_B(j)})}{(x_{\pi_B(s)} - x_{\pi_B(j)})} (\mathrm{mod} p)$$

여기에서 x_i는 공개된 값이며 변형된 분산정보, $a_{\pi_B(s)}$는 다음과 같이 정의한다.

$$a_{\pi_B(s)} = K_{\pi_B(s)} \prod_{j=1, j\neq s}^{t} \frac{(0 - x_{\pi_B(j)})}{(x_{\pi_B(s)} - x_{\pi_B(j)})} (\mathrm{mod} p)$$

변형된 분산정보 $a_{\pi_B(s)}$를 다른 사람에게 전달하는 것은 분산정보 $K_{\pi_B(s)}$를 전달하는 것과 동일한 효과를 갖는다. 그러므로 각 참가자의 변형된 분산정보는 실제 분산정보와 동일한 보안성을 유지해야 한다.

나. ElGamal 암호시스템

ElGamal 암호시스템[9]을 사용하기 위해서는 유한체(finite field) F_p상에서 발생자(generator) g를 선택한다. 비밀 관리자는 $0 < a < p-1$ 범위에서 임의의 정수값 a를 발생시킨다. 비밀 관리자는 a의 복사본을 파괴하기 전에 회사에 비밀키로서 a를 제공하고 g^a를 공개키로 공개한다. 메시지 M을 전송하기 위하여 전송자는 임의의 정수값 k를 생성하고 암호문 $C = (g^k, Mg^{ak})$를 전송한다. 이 메시지를 복원하기 위해서 수신자는 암호문 첫 번째 인자인 g^k를 a승 해주기만 하면 된다. 이 결과 값의 곱셈에 대한 역수인 g^{-ak}를 Mg^{ak}와 곱해주면 메시지 M을 복원하게 된다.

다. 변형된 분산정보 이용 공개키 인증 과정

전송자는 t명의 참가자들에게 메시지를 전송하고 각 참가자는 메시지를 암호화하여 분산 저장 후 필요시에 이 정보를 이용하여 메시지를 복호하려고 한다. 수신자는 t명의 참가자가 변형된 분산정보와 ElGamal 암호문을 이용하여 생성한 부분결과를 전송받아서 원래의 메시지를 복원한다.

우선 ElGamal 암호시스템 초기 설정 단계에서 비밀 관리자는 비밀 키 a를 선정한 후, 각 참가자에게 분산정보 a_i를 할당한다. 암호화 과정은 ElGamal 암호시스템과 동일하며, 복호화 과정은 다음과 같은 절차를 통해 수행된다.

메시지가 도착하면 t명의 참가자 $\pi_B(s)$로 이루어진 부분집합 B는 3.1.2의 (가)에서와 같이 변형된 분산정보 $a_{\pi_B(s)}$를 계산한다. 이렇게 계산된 변형된 분산정보의 합은 비밀 키 $a \bmod \phi(p)$와 합동이다.

각 참가자 $\pi_B(s)$는 g^k에 $-a_{\pi_B(s)}$승을 하여 각각의 부분결과 $g'_{\pi_B(s)}$를 얻는다. 이 부분결과 값이 메시지를 복원할 수신자에 전송된다.

수신자는 메시지 M을 얻기 위해서 모든 $g'_{\pi_B(s)}$를 곱한다. 이 결과 값인 g^{-ak}를 암호문의 2번째 인자인 Mg^{ak}에 곱하여 메시지 M을 얻게 된다.

각 노드의 분산정보 $K_{\pi_B(s)}$ 대신에 $g^{K_{\pi_B(s)}}$를 전송함으로써 분산정보 값이 공개되는 것을 방지하면서도 공개키를 인증할 수 있게 된다. 이 경우 k값이 항상 일정해도 무방하므로 $k=1$로 설정하고 공개키 인증에 사용되는 $g^{K_{\pi_B(s)}}$를 지수 분산정보라고 정의한다.

3.1.3 지수분산정보 이용 공개키 및 노드 인증

가. 센서노드의 인증정보 생성 및 저장

네트워크 관리자는 센서 노드를 필드에 배치하기 이전에 다음과 같은 과정을 통하여 각 센서노드에 공개키 쌍(pk_i, sk_i), 지수 비밀(g^K), 지수 분산정보(K_i')를 생성하여 저장한다. 그리고 기본적으로 발생자(g), 분산정보 보다 충분히 큰 소수(p), 임계치(t), ID(센서 식별자) 등 연산에 필요한 정보도 생성한다.

① 각 네트워크 관리자는 ECC 등의 공개키 방식을 이용하여 각 센서 노드에 공개키 쌍(pk_i, sk_i)을 할당한다. 그리고 다음과 같이 임의의 $t-1$차 다항식을 생성한다.

$$F(x) \equiv ax^{t-1} + bx^{t-2} + cx^{t-3} + \cdots + K \bmod p$$

② 각 센서 노드의 분산정보(K_i)는 다음과 같은 식을 이용하여 생성하고

$$K_i = F(pk_i)$$

지수 분산정보(K_i')를 다음과 같이 계산한다.

$$K_i' = g^{K_i}$$

나. 공개키 및 노드 인증

필드에 배치된 센서 노드는 다음과 같은 과정을 통하여 이웃 센서 노드의 공개키를 인증함으로써 해당 노드를 인증한다.

① 각 센서 노드는 이웃 센서 노드에 자기의 공개키 값(pk_i), 지수 분산정보(K_i'), ID를 포함하는 인증정보를 브로드캐스팅 한다.

② 각 센서 노드는 이웃 노드에서 $t-1$개 이상의 인증정보를 수신하고, 센서 노드 자신과 이웃노드의 $t-1$개의 $(pk_{\pi_B(s)}, K'_{\pi_B(s)})$ 부분집합 B를 설정$(\pi_B : B \rightarrow \{1,2,\cdots,n\}, |B| = t)$한다.

③ 각각의 $K'_{\pi_B(s)}$에 $\prod_{j=1, j\neq s}^{t} \frac{(0 - pk_{\pi_B(j)})}{(pk_{\pi_B(s)} - pk_{\pi_B(j)})}$승을 하여 부분결과 $g^{a_{\pi_B(s)}}$를 얻는다.

④ 아래 식과 같이 t개의 부분결과를 모두 곱하여 얻은 결과가 지수 비밀와 동일하면 $t-1$개의 센서 노드의 공개키를 동시에 인증함과 동시에 해당 노드를 인증함으로써 t개의 노드들이 동시에 신뢰관계를 형성하게 된다.

$$\prod_{s=1}^{t} g^{a_{\pi_B(s)}} = g^K$$

이웃 노드 중에서 최소한 $t-1$개 이상이 정상이라면 위와 같은 방법을 통하여 정상적인 이웃노드를 모두 인증할 수 있게 된다. 이와 같이 과정을 거쳐 이웃 노드를 모두 인증한 후에는 인증한 이웃 노드의 인증정보에 포함된 공개키로 데이터를 암호화하여 송신하면 그 이웃 노드는 자신의 비밀키로 데이터를 복호함으로써 이웃 노드간 암호화통신이 가능하게 된다. 또한 각 노드가 브로드캐스팅한 지수 분산정보만으로는 원래의 분산정보를 알아낼 수가 없으므로 인증정보를 위조할 수가 없게 되며, 인증과정에서 사용된 인증정보에 대한 기밀성이 보장된다.

3.2 *Hash*값을 이용한 악의적 행위 탐지 기법

본 연구에서 제안된 노드 인증기법에 사용되는 정보는 공개키 알고리즘에서 사용되는 공개키 pk_i, 모든 센서 노드가 동일하게 가지고 있는 지수비밀 g^K, 그리고 지수 분산비밀 K_i' 3가지이다. 그 중 지수 분산정보는 각 노드의 공개키를 이용하여 생성하는 정보이기 때문에 공개키와 동일한 의미를 가진다. 따라서 정확한 인증정

보의 제공 여부를 검증함으로써 인증과정을 방해하는 악의적인 공격행위를 탐지하기 위해서 각 노드에 저장된 공개키(pk_i)와 지수 비밀(g^K)정보만을 이용한다.

지수 비밀 g^K의 경우 모든 센서 노드들이 동일한 값을 보유하고 있으며, 필드 배치 이전에 안전하게 저장되는 값일 뿐만 아니라 이산대수문제의 어려움을 기반으로 생성된 값이기 때문에 공격자가 임으로 생성할 수 없는 값이다. 그리고 공개키(pk_i) 값 은 모든 센서 노드들에게 공개된 값이므로 알 수 있다. 따라서 모든 센서 노드는 상대방이 전송한 공개키(pk_i) 값과 자신이 보유하고 있는 지수 비밀 g^K를 연결(concatenation)하여 $Hash$ 값을 구하고, 이를 상대방이 생성하여 전송해 준 $Hash$ 값과 비교해 보면 손쉽게 정당한 인증정보를 전송하였는지의 여부를 검증할 수 있게 된다. 구체적인 탐지 과정은 다음과 같다.

① 모든 센서 노드는 필드에 배치되기 전에 저장한 공개키 pk_i와 지수비밀 g^K를 연결하고 $Hash$함수를 이용하여 H_K를 추가적으로 메모리에 저장한다.

$$H_K = Hash(pk_i \parallel g^K)$$

② 각 센서 노드는 인증정보를 브로드캐스트하는 과정에서 생성된 H_K도 함께 브로드캐스트한다.

③ 인증에 필요한 정보와 H_K를 전송받은 각 센서 노드들은 인증을 위한 다항식을 계산하기 이전에 먼저 H_K의 유효성을 검증한다. 이를 위해 각 노드는 인증에 참여하는 $t-1$개의 센서 노드들을 대상으로 해당 노드가 전송한 공개키 pk_i와 자신이 가지고 있는 지수 비밀 g^K를 이용하여 H'_K를 생성한다.

④ 인증에 참여하는 각 센서 노드에 대해 자신이 계산한 H'_K와 전송받은 H_K값을 비교한다.

⑤ 특정 센서 노드의 H_K정보가 자신이 계산된 값과 일치하지 않는다면 인증과정에서 배제시키기 위해 해당 센서 노드의 ID를 네트워크 내로 브로드캐스트하여 인접 센서 노드들에게 알려주고, 인접한 센서 노드들 중 다른 센서 노드를 선택하여 인증작업을 계속 수행한다. 이상과 같은 과정을 거치지 않고, 인증과정에서의 다항식 계산이 수행된 이후에 잘못된 인증정보를 전송한 노드를 발견하고자 할 경우에는 어떤 센서 노드가 잘못된 인증정보를 전송함으로써 인증을 방해하고 있는지를 파악하는 것이 불가능함은 물론 인증을 방해하는 센서 노드가 1개인지 아니면 그 이상인지도 파악할 수 없어 최악의 경우 일대일 인증에서보다 훨씬 더 많은 자원이 소모될 수도 있다. 그러나 다항식 계산 이전에 비교적 연산효율성이 보장되는 $Hash$함수와 이미 알고 있는 공개키(pk_i) 값 및 지수 비밀(g^K) 정보를 이용하여 인증정보의 유효성 여부를 검증함으로써 인증 방해 행위를 조기에 탐지해 낸다. 이러한 기법은 악의적인 인증 방해 공격 탐지에 소요되는 자원소모를 효율적으로 줄이면서도 조기에 탐지해 낼 수 있는 장점을 가진다.

Ⅳ. 성능분석

4.1 자원 효율성 분석

4.1.1 메모리 사용량 분석

전체 네트워크의 노드 수가 n이면 (t,n)임계치 기법을 이용하여 공개키 인증을 할 때 한 번의 인증 작업에 필요한 이웃노드의 수는 $t-1$이다. Merkle 인증 트리 기반의 공개키 인증방식에서는 한 노드가 이웃 노드를 인증하기 위해서 사용하는 메모리 공간은 $(\log_2 n)+1$이다. 이에 반해 본 연구에서 제안된 인증방식의 메모리 사용 공간은 3으로 항상 일정하다. 즉 모든 센서 노드가 공유하는 지수 비밀, 센서 노드 각각이 할당받은 지수 분산 정보, 그리고 유효한 인증정보를 보유하고 있는지의 여부를 검증하기 위한 공개키와 지수 비밀의 해시값만 저장하고 있으면 된다. 제안하는 공개키 인증방식은 인증 시 필요한 각 센서의 메모리 공간이 Merkle 인증 트리 기반 공개키 인증방식의 $O(\log_2 n)$에서 상수로 줄어든 것을 알 수 있다.

그림 2는 제안하는 인증 기법과 Merkle 트리 방식을 메모리 사용량 측면에서 비교한 결과이다. 제안된 인증기법은 네트워크의 크기에 상관없이 일정한 메모리 용량만을 필요로 하고 있기 때문에 Merkle 트리방식에 비해 엄청난 성능 향상을 보이고 있다. 즉 소수 p값을 충분히 크게 해주면 네트워크의 크기에 상관없이 인증에 필요한 메모리 공간의 크기가 일정함을 알 수 있다. 물론

인증작업을 수행하는 과정에서는 주변 센서 노드의 지수 비밀과 지수 분산정보, 공개키와 지수 비밀의 *Hash* 값을 저장하는 것이 필요하며, 이 경우에는 $3\times t$의 메모리 공간이 추가로 요구된다.

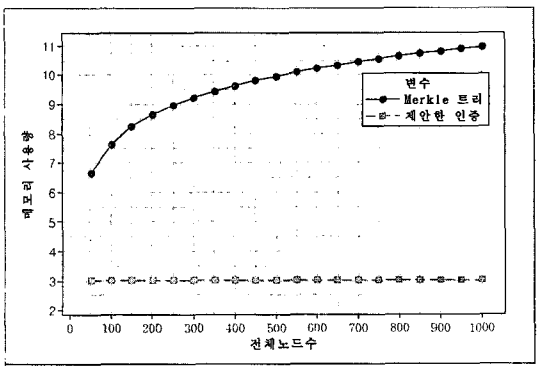

그림 2. 메모리 사용량 비교
Fig. 2 Memory usage comparison

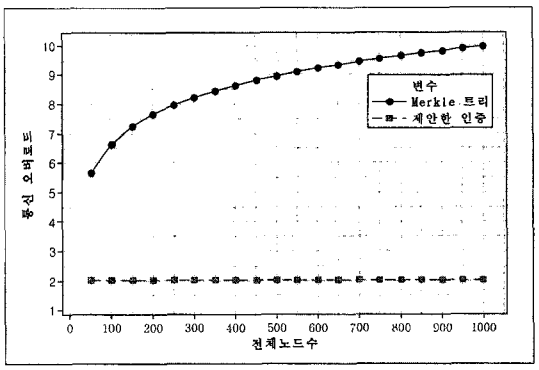

그림 3. 통신 오버로드 비교
Fig. 3 Communication overload comparison

4.1.2 통신 오버로드 분석

각 센서 노드는 필드에 배치된 후에 이웃 노드와의 상호 인증을 위하여 ID, 공개키, 분산정보, 해시값 등으로 이루어진 인증정보를 교환한다. Merkle 트리 기반의 인증방식에서 공개키 인증을 위해 필요한 통신 오버로드는 $\log_2 n$이다. 즉, Merkle 트리 기반의 방식에서 각 센서 노드는 자신의 공개키 이외에, 필드에 배치되기 전에 공개키 인증을 위해 저장한 경로 λ상의 형제노드의 Φ값을 전송해야만 한다. 이에 반해 제안한 인증방식의 통신 오버로드는 2로 항상 일정하다. 즉 제안한 인증방식에서 각 센서 노드는 자신의 공개키 이외에, 필드 배치되기 전에 할당받은 지수 분산정보와 해시값만 이웃 노드에 브로드캐스팅하면 되는 것이다.

결과적으로 공개키 인증방식은 인증시 소요되는 통신 오버로드가 Merkle 트리 기반 방식의 $O(\log_2 n)$에서 상수로 줄어드는 것을 알 수 있다. 이는 네트워크의 크기에 상관없이 일정한 통신 오버로드만이 필요한 것으로 기존 방식들에 비해 엄청난 성능 향상을 이룬 것이다. 즉 모든 계산에 사용되는 공통 모듈러 소수 p값을 충분히 크게 해주면 네트워크의 크기에 상관없이 인증에 사용되는 통신오버로드의 크기가 일정하게 유지된다.

그림 3은 제안하는 인증 기법과 Merkle 인증 트리 기반의 공개키 인증 기법을 통신 오버로드 측면에서 비교한 결과이다. 그림에서 알 수 있는 바와 같이 제안한 인증방식의 통신 오버로드는 네트워크의 크기가 커질수록 Merkle 트리 기반의 인증방식보다 상대적으로 효율적이다.

4.1.3 연산성능 분석

Merkle 트리 기반의 인증방식에서 각 센서 노드는 이웃 센서 노드에 대해 한 개씩 순차적으로 인증작업을 수행하는 반면, 제안한 인증방식에서는 $t-1$개의 이웃 노드를 동시에 인증한다. 두 방식에서 한 번 인증작업을 수행하는 데에 동일한 계산량이 소요된다면 제안한 인증방식이 최고 $t-1$배 연산 성능이 우수하다고 할 수 있다. 특정 센서 노드의 이웃 노드 밀도가 d이고 오류 분산정보가 포함되어 있지 않았을 경우 $d\%(t-1)$회의 인증작업만으로 주변 센서 노드의 모든 공개키를 인증하고 신뢰관계를 형성할 수 있게 된다.

그러나 인증작업에 관여한 분산정보에 오류가 포함되어 있을 경우를 대비해 이를 사전에 제거해 내기 위해서는 추가적인 연산 작업이 요구되긴 하지만, 3.2절에서 설명한 바와 같이 공개키(pk_i)와 지수 비밀(g^{f_i})에 대한 *Hash*값만으로 간단하게 제거가 가능하기 때문에 연산 성능이 크게 저하되지는 않을 것으로 판단된다.

4.2 보안성 분석

4.2.1 인증서 위조 가능성 분석

우선 공격자가 목표 네트워크에서 공개키 인증을 위하여 Shamir의 보간 다항식을 이용한 (t,n)임계치 기법을 사용한다는 것을 알고 있다고 가정한다. 또한 공격자

는 t개의 지수 분산정보를 감청하였고 연산에 사용되는 모듈러 소수 p, 생성자 g를 알고 있다고 했을 때, 공격자는 이러한 정보를 이용하여 모든 노드가 공유하고 있는 인증을 위한 비밀까지는 쉽게 계산할 수 있을 것이다.

공격자는 자신이 정상적인 노드인 것처럼 위장하여 네트워크에서 인증을 받기 위해서는 자기의 공개키가 포함된 인증서가 주변노드에서 인증을 받도록 해야 할 것이다. 이를 위해서는 자기의 공개키와 연결된 지수 분산정보를 생성하여야만 한다. 하지만 이는 Shamir의 보간 다항식을 아는 자만이 만들 수 있고 이 다항식을 생성하기 위하여 t개의 분산정보를 알아야 한다. 그러므로 공격자가 지수 분산정보에서 원래의 분산정보를 계산해 낼 수 있다면 그 다항식을 계산해 내고 그의 공개키 값을 대입하여 그만의 지수 분산정보를 위조할 수 있는 것이다.

분산정보가 K일 때 지수 분산정보는 g^{K_i}이므로 g와 g^{K_i}를 알고 있을 때 K를 계산하는 것은 이산대수문제를 해결하는 것과 동일한 난이도를 갖는 것으로서 매우 어려운 일이다.

4.2.2 노드 포획시 보안성 분석

어느 센서 노드가 공격자에 의해 포획되면 센서 내에 저장된 공개키 쌍 등이 노출된다. 이러한 경우에는 3.2절에서 언급한 바와 같이 유효한 인증정보가 아닌 잘못된 인증정보를 전송했을 경우에는 $Hash(pk_i \| g^k)$ 탐지 기법을 적용하면 쉽게 탐지해 낼 수 있다.

4.3 기타

제안한 방식은 Merkle 트리 공개키 인증 방식에 비하여 확장성이 뛰어나다. Merkle 트리 방식에서는 네트워크의 크기가 정해지고 인증정보가 계산된 이후에는 네트워크 크기를 늘리게 불가능하다. 하지만 제안한 방식의 네트워크의 크기는 소수의 크기로 제한되지만 실제로는 아주 큰 값으로 설정되어 있으므로 네트워크 크기에 무관하게 인증정보를 계속적으로 생성할 수 있다.

제안한 방식에서 (t,n)임계치 기법을 이용하여 공개키 인증할 때 t는 노드의 밀집도 d보다 충분히 작게 설정해야 한다. 만약에 이웃노드의 수가 부족해서 인증작업을 하지 못하는 경우가 발생하면 전체 네트워크의 연결성을 보장할 수 없는 것이다. 물론 필드의 변두리에 배치된 센서노드는 밀집도가 필드 중심부의 노드보다 낮을 확률이 크므로, 변두리 센서노드와 이의 이웃센서 노드는 필요에 따라 인증작업 동안 송신 출력을 높일 필요가 있을 것이다. 변두리 센서노드는 중심부의 센서노드보다 메시지를 중계하는데 소모되는 전원이 적으므로, 인증작업 동안 더 많은 전원을 소모하는 것은 큰 문제가 되지 않을 것이다. 다른 해결 방안으로는 센서노드 1개에 여러 개의 인증정보를 저장하게 할 수도 있지만 이것은 메모리의 효율성을 떨어지게 하므로 몇 개의 인증정보를 1개의 센서 노드에 저장하는 게 적당한 지에 대한 최적화 방안도 연구되어야 할 것이다. 성능 비교 · 분석 결과를 표 1로 요약하였으며 제안한 인증방식이 Merkle 트리 공개키 인증방식보다 연결성을 제외한 나머지 분야에서 우수한 특성을 보이는 것을 알 수 있다.

표 1. 공개키 인증성능 비교
Table. 1 Public key authentication performance comparison

성능 \ 인증방식	Merkle 트리 인증방식	제안한 인증방식
메모리	$\log_2 n+1$	3(g^K, K_i', H_K)
통신 오버로드	$\log_2 n$	2(K_i', H_K)
인증연산 (노드당)	d <hashing>	$d\%(t-1)$ <exponentiation, multiplication>
확장성	제한	보장
연결성	보장	$t-1$개 이상의 이웃노드 필요

V. 결 론

본 연구에서는 WSN에서 효과적으로 사용될 수 있는 노드 인증 기법으로 비밀분산기법의 일종인 (t,n) 임계치 기법과 변형된 지수 비밀정보를 이용한 공개키 및 노드 인증방식을 제안하였다.

제안한 인증방식은 각 노드가 필드에 배치되기 전에 저장하는 공개키 정보와 비밀분산기법의 적용을 위해 요구되는 일부 정보들로 한정함으로써 네트워크 크기에 따라 메모리 사용량이 증가하는 문제점을 해결하였으며, 동시다발적 인증을 실시함으로써 인증에 소요되

는 시간을 최소화하였다. 확장성을 지원하기 위해 Merkle 트리 기반의 인증처럼 사전에 정해진 정보를 기반으로 인증을 실시하지 않고, 연결성이 보장되는 노드들의 수만 충족된다면 언제든 주변 노드들에 대한 인증이 가능하도록 비밀분산기법을 응용하여 인증에 적용하였으며, 기밀성을 보장하기 위해 각 노드에 저장되고 인증에 사용되는 분산비밀정보를 ElGamal 암호시스템을 적용하여 지수화 하였다. 또한, 잘못된 인증정보를 제공한다면 정상적인 노드 인증이 이루어질 수 없는 문제점을 인식하고 잘못된 인증정보를 이용, 인증을 방해하는 공격행위를 조기에 발견할 수 있는 메커니즘을 추가적으로 제안하였다.

본 연구에서 제안한 인증방식은 WSN에서 초기에 동시 다발적인 대규모 인증이 필요한 경우 기존의 어느 공개키 인증방식보다 유용하게 사용될 수 있는 주요 보안기술이 될 것으로 판단된다.

참고문헌

[1] G. Gaubatz, J. Kaps, and B. Sunar, "Public keys cryptography in sensor networks - revisited", In The Proceedings of the 1st European Workshop on Security in Ad-Hoc and Sensor Networks (ESAS), 2004

[2] Srdjan Capkun, Levente Buttyan and Jean-Pierre Hubaux, "Small Worlds in Security Systems; an Analysis of the PGP Certificate Graph", In Proceedings of the ACM New Security Paradigms Workshop 2002, pp.2, 2002.

[3] Dirk Balfanz, D. K. Smetters, Paul Stewart and H. Chi Wong, "Talking To Stranger: Authentication in Ad-Hoc Wireless Networks", In Proceedings of the Network and Distributed System Security Symposium 2002, 2002.

[4] Wenliang Du, Ronghua Wang, and Peng Ning, "An Efficient Scheme for Authenticating Public Keys in Sensor Networks", 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), 2005.

[5] Y. Desmedt. "Society and group oriented cryptography : a new concept", In C. Pomerance, editor, *Advances in cryptology, Proc. of Crypto'87(Lecture Notes in computer Science 293)*, pages 120-127. Springer-Verlag, 1988. Santa Barbara, california, U.S.A., August 16-20

[6] A. Shamir, "How to share a secret", Commun. ACM, 22:612-613, November 1979

[7] T. El Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", IEEE Trans. Inform. Theory, 31:469-472, 1985

[8] Y. Desmedt and Y. Frankel, "Threshold cryptosystems", in Advances in Cryptology - Crypto '89, Proceedings, Lecture Notes in Computer Science 435, G. Brassard, Ed., Santa Barbara: Springer-Verlag, 1990, pp. 307-315

[9] W. Diffie and M. E. Hellman, "New directions in cryptography", *IEEE Trans. Inform. Theory*, IT-22(6); 644-654, November 1976

저자소개

김일도(Il-do Kim)

고려대학교 전산학과 박사
해군사관학교 교수

※ 관심분야 : 데이터베이스 보안, NCW 보안, 센서네트워크 보안

김동천(Dong-cheon Kim)

국방대학교 전산정보학과 석사
해군사관학교 전임강사

※ 관심분야 : 침입탐지 시스템, 노드 인증