무선 센서 네트워크에서 클러스터 헤더를 통한 오용키 검출을 위한 검증 방법 †

박민우*, 김종명*, 한영주**, 정태명***
*성균관대학교 전기전자컴퓨터공학과
**성균관대학교 컴퓨터공학과
***성균관대학교 정보통신공학부
e-mail : {mwpark, jmkim, yjhan}@imtl.skku.ac.kr, tmchung@ece.skku.ac.kr

Misused key detection at cluster header in wireless sensor network

Min-Woo Park*, Jong-Myoung Kim*, Young-Ju Han**, Tai-Myoung Chung***
*Dept. of Electrical and Computer Engineering, Sungkyunkwan University
**Dept. of Computer Engineering, Sungkyunkwan University
***School of Information Communication Engineering, Sungkyunkwan University

요 약

무선 센서 네트워크는 주변 정보를 감지할 수 있는 다수의 센서들로 구성된 네트워크로 다양한 분야에서 활용되고 있다. 과거에는 무선 센서 네트워크 환경에서 각 센서들 간의 비밀통신이 중요한 이슈였으며, 특히 이를 위한 키 관리 기법들이 주요 연구방향이었다. 하지만 잘 분배되고 관리된 키라 할지라도, 공격자에 의해 특정한 센서 노드(node)가 수집되면, 노출된 노드(compromised node)가 가지고 있는 키가 공격자에게 들어나게 된다. 노출된 공유키(shared key)를 통해 노출되지 않은 정상 노드(non-compromised node) 사이의 대칭키(pairwise key)를 얻을 수 있으며 결국 공격자는 네트워크에 심각한 영향을 줄 수 있는 메시지 삽입 및 수정 공격을 감행할 수 있다. 본 논문에서는 이와 같은 공격을 탐지하고 오용된 키(misused key)를 폐기하기 위한 방법으로 DAC(detection at cluster header) 기법을 제안한다.

1. 서론

최근 무선 센서 네트워크는 하드웨어 기술이나, 소프트웨어 알고리즘 등 다방면으로 눈부신 발전을 거듭해왔다. 무선 센서 네트워크는 제한된 자원과 에너지를 가진 수많은 센서들로 이루어지며, 이러한 센서들은 주변 환경으로부터 정보를 수집하며 이를 BS(Base Station)에 전달한다. 최근 이러한 무선 센서 네트워크는 군사, 학문, 기업, 관측 등 다양한 분야에서 널리 쓰이고 있다.

무선 센서 네트워크에서 보안 서비스를 제공하기 위해 많은 연구가 진행되어 왔다. 대부분 센서 노드의 제한된 자원과 에너지로 인해 많은 계산 능력을 필요로 하는 공개키(public key) 암호 방식보다 대칭키(secret key) 암호 방식이 주로 사용된다. 하지만 대칭키 암호 방식을 사용하기 위해서는 각각의 센서 노드 간에 유일한 키(unique key)를 나눠가져야 하는데, 무선 센서 네트워크를 구성하는 센서들의 수와 센서 노드의 제한된 저장공간으로 인해 불가능 하다. 따라서 무선 센서 네트워크에서는 대칭키 암호 방식을 사용하되 각각의 노드가 유일한 키를 나눠가지지 않고 공유키(shared key)를 소유하며 이를 통해 다른 노드와 비밀 통신을 수행한다[1]. 이 때 공유키는 유일한 키가 아니라 다른 노드들과 중복해서 소지하는 키이다. 센서 노드는 통신하고자 하는 노드와 직접적으로 같은 공유키를 소지하고 있다면 그 키를 통해 대칭키(pairwise key)를 설정하고, 그렇지 않다면 제 3 의 노드를 통해 대칭키를 분배한다.

무선 센서 네트워크에서 공격자는 센서 노드에대해 물리적 획득 공격을 감행할 수 있으며 공격자는 획득된 노드를 통해 쉽게 공유키들을 수집할 수 있다. 공격자는 수집된 키를 통해 노출되지 않은 정상적인 노드(non-compromised node)들 간의 통신 내용을 도청할 수 있으며, 심지어 임의로 메시지를 수정 하거나 삽입 할 수 있다. 특히 후자의 경우 센서네트워크의 어플리케이션의 동작에 심각한 문제를 야기할 수 있다. 우리는 여기서 공격자가 정상 노드 사이의 대칭키를 획득하여 공격을 감행한 경우 이 키를 오용키(misused key)로 정의한다.

오용키를 탐지하기 위한 기법으로 분산 검증 모델(distributed detection scheme)[2]이 있다. 분산 검증 모델은 메시지를 수신하는 노드가 검증을 위해 검증 메시

† "본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음" (IITA-2008-C1090-0801-0028)

지를 다른 노드에게 전송한다. 이와 같은 분산 검증 모델은 계층적 라우팅 구조를 고려하지 않은 검증 방법으로, 계층적 라우팅 구조의 무선 센서 네트워크 환경에서는 수신 노드, 즉 클러스터 헤더(cluster header)에 부하(overhead)가 집중되어 클러스터 헤더의 자원을 크게 소모하는 문제가 있다.

본 논문에서는 이러한 분산 검증 모델의 문제점을 해결하기 위한 새로운 검증 모델인 DAC(detection at cluster header)를 제안한다. DAC 모델은 수신 노드가 검증을 위한 추가적인 메시지 전송하지 않기 때문에 기존의 분산 검증 모델에 비해 경제적인데, 이러한 특징은 지속적으로 여러 노드로부터 데이터를 수신하는 클러스터 헤더에서 두드러지게 나타난다. 또한 기존의 분산 검증 모델에서는 검증 노드가 검증 메시지를 한 홉으로 BS 에 전송하여 검증 노드에 큰 부하가 가해졌던 반면에 DAC 모델은 BS 과 통신을 수행하는 클러스터 헤더가 검증 메시지를 추가하여 전달함으로써 검증 노드의 부하 역시 효과적으로 줄이는 모델이다.

본 논문의 구성은 다음과 같다. 2 장에서 Liu 가 제안한 분산 검증 모델에 대해 살펴보며, 3 장에서는 DAC 모델을 소개한다. 4 장에선 결론과 향후 연구 방향에 대해 서술한다.

2. 오용키 검증 기법

오용키를 검증 하기 위해 Liu 는 세가지 검증 기법을 제안하였다[2]. 세가지 기법은 모두 공통적으로 검증 메시지(Committing message)를 생성하는데, 검증 메시지(V)는 메시지를 전송한 센서와 메시지 내용 둘 모두를 인증 가능한 정보를 포함한다. 따라서 검증 노드 혹은 BS 에서 검증 메시지를 통해 오용키를 이용한 공격을 검증 할 수 있다..

먼저, 첫 번째 검증 기법은 BS 에서 검증을 수행하며, 검증 메시지에는 단일 메시지(M)에 대한 인증 정보만 포함되어있다. 따라서 첫 번째 검증 기법은 하나의 검증 메시지당 하나의 메시지만 검증이 가능하다. 이때 검증 메시지는 송신 노드(u)와 BS 사이의 비밀키(K_u)를 통해 생성 된다. 수신 노드(v)는 일정한 확률로 검증 메시지(V)와 메시지(M)을 BS 에 보내어 검증을 요청하며, BS 는 송신 노드의 비밀키(K_u)를 통해 이를 검증한다.

두 번째 검증 기법의 검증 메시지에는 두 노드가 통신을 시작한 이후 송신한 전체 메시지에 대한 인증 정보가 포함된다. 이러한 인증 정보는 검증값(commit value, $C_{u,v}$)라 하며, 송신 노드(u)와 수신 노드(v)에 의해 각각 따로 관리된다. 검증값($C_{u,v}$)는 초기에 0 으로 설정되며, 송신 노드가 메시지 M 을 전송할 때 마다 해쉬 함수를 통해 $H(C_{u,v} \,||M)$로 갱신한다. 수신 노드 역시 수신한 메시지 M 을 통해 새로운 검증값($C_{u,v}$)을 계산하여 갱신한다. 따라서 오용키를 통한 메시지 공격이 없다면 두 검증값은 항상 같은 값을 유지하며, 수신 노드는 전달받은 검증 메시지(V)와 자신의 검증값을 BS 에 전달하며, 기지국에서는 두 노드의 검증값이 동일한지 확인 함으로써 오용키를 검증한다.

마지막 검증 기법은 분산 검증 기법이다. 이 기법은 무선 센서 네트워크에 검증 노드를 추가적으로 두고 검증 노드를 통해 오용키를 찾아내는 기법이다. 추가된 검증 노드들은 각각의 센서 노드들과 각각 유일한 키를 나눠가지는데, 이러한 유일한 키는 각각의 센서 노드가 BS 와 나눠가진 비밀키(K_u)와 해당 검증 노드를 구별하기 위한 ID(i)의 해쉬 값 $H(K_u||i)$을 통해 생성된다. 따라서 각각의 센서 노드는 검증 노드들과의 유일한 키를 저장할 필요 없이 검증 받고자 하는 검증 노드가 정해지면 해당 키를 간단한 연산을 통해 구할 수 있다. 송신 노드는 이전의 두 검증 기법들과 달리 두 개의 검증 메시지(V_1, V_2)를 생성한다. V_1 는 검증 노드에서 검증 받기 위한 메시지이며, V_2 는 BS 에서 검증 받기 위한 메시지이다. 검증 노드에서 오용키를 탐지하더라도 즉시 해당 오용키를 센서 노드들에게 알려 배제시키지 않고 BS 에서 한번 더 검증을 받는다. 이를 위해 검증 노드는 수신 노드로부터 받은 검증 메시지($u,v,C_{u,v},V_2$)를 BS 에 전송한다. 이는 검증 노드 역시 물리적 획득 공격에 취약하기 때문에 오용키에 대한 처리는 오직 BS 에서만 수행하기 위해서다.

앞서 살펴본 세가지 검증 기법은 모두 수신 노드(v)가 검증 노드 혹은 BS 에게 검증 메시지를 전송하여 검증 받는다. 이러한 검증 과정은 계층적 라우팅을 사용하는 무선 센서 네트워크에는 부적합하다. 클러스터를 통해 하나의 노드가 여러 센서 노드로부터 데이터를 모아 BS 에 전송하는 계층적 라우팅에서는 클러스터 헤더가 검증 과정을 전담하기 때문에 클러스터 헤더에 부하가 집중되며, 그로 인한 극심한 에너지 손실을 겪게 된다. 또한, 기존의 방법들은 검증 메시지를 직접 BS 에 전송하는데 이 경우 각각의 검증 노드에게 큰 부하로 작용한다.

3. DAC 기법

이 장에서는 계층적 라우팅을 사용하는 무선 센서 네트워크에서 오용키를 검증하기 위한 효과적인 검증 기법인 DAC 기법을 설명한다.

각 센서 노드들은 BS 와 둘만이 나눠가지는 비밀키를 가지며, BS 는 비밀키를 통해 센서 노드를 인증할 수 있다. DAC 기법은 두 가지 검증 방법을 병행하는데, 하나는 수신 노드가 직접 검증을 하는 방법이고, 다른 하나는 BS 에서 검증을 하는 방법이다. BS 에서 검증은 센서의 비밀키를 통해 만들어진 메시지(V)를 통해 이루어 지며, 수신 노드에서 검증은 송신 노드와 수신 노드의 누적 검증값(cumulative commitment value)을 통해 이루어진다. 각각의 방법에 대해 자세히 살펴 보겠다.

3.1 수신 노드의 검증 과정

DAC 기법은 무선 센서 네트워크에 특수한 역할을 하는 안전 노드(safe node)를 추가하여 사용한다. 안전 노드는 모든 노드들과 유일한 키를 공유한다. 이때 안전 노드와 각 노드들 간의 유일한 키 K_{u,S_i} 는 (수식

1)과 같이 구해진다. (수식 1)의 $H()$는 해쉬 함수를 의미하고, i는 안전 노드의 ID, K_u는 각 노드와 BS가 공유하고 있는 비밀키를 나타낸다. 즉, 안전 노드와 노드들 간의 유일한 키는 간단한 연산을 통해서 구해진다. 따라서 각 센서 노드들은 추가적인 메모리 낭비 없이 안전 노드와 유일한 키를 공유할 수 있다.

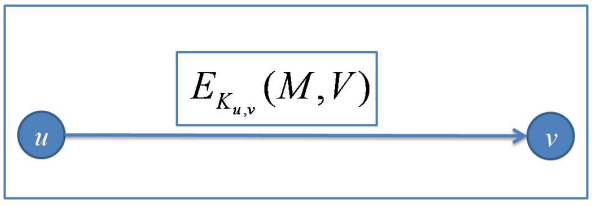
$$K_{u,S_i} = H(K_u \| i) \qquad (1)$$

안전 노드는 센서 노드들과 유일한 키를 가진다는 점에서 분산 검증 기법의 검증 노드와 유사하지만, 그 기능은 전혀 다르다. DAC 기법에서 안전 노드는 단지 유일한 키를 통해 송신 노드(u)와 수신 노드(v)를 연결하는 안전한 통로로만 사용된다.

수신 노드와 송신 노드는 각자 검증값($C_{u,v}$)를 각각 따로 관리한다. 검증값은 (수식 2)와 같이 구해지며, 두 식 중 위의 식은 초기화를 나타내며 아래 식은 검증값의 갱신 과정을 나타낸다. M은 전송하는 메시지를 의미한다. 수신 노드와 송신 노드는 각자 따로 검증값을 관리하지만 노출된 노드에 의해 메시지가 삽입 되지 않는 이상 송신 노드와 수신 노드는 같은 검증값을 유지한다. 따라서 두 노드의 검증값 비교를 통해 오용키를 검증할 수 있다.

$$\begin{cases} C_{u,v} = 0 \\ C_{u,v} = H(C_{u,v} \| M) \end{cases} \qquad (2)$$

수신 노드에서 검증값을 비교하기 위해 송신 노드의 검증값을 필요로 한다. 따라서 송신 노드는 일정한 주기로 안전 노드를 통해 수신 노드에게 검증값을 안전하게 전달한다. 이때 검증값을 전달하는 주기는 수신 노드가 결정하며, 수신 노드가 통신을 시작할 때 수신 노드에 인접한 안전 노드 목록과 함께 알려준다. 수신 노드의 안전 노드 목록은 송신 노드가 검증값을 전송할 때 사용할 안전 노드의 목록을 의미한다.



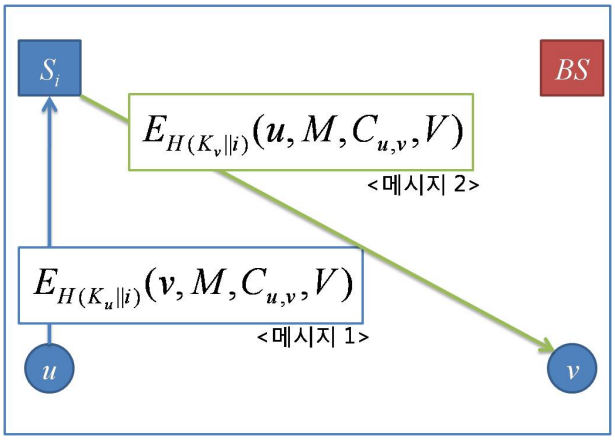
(그림 1) 일반적인 데이터 전송

(그림 1)은 일반적인 데이터 전송을 나타낸다. 송신 노드(u)는 수신 노드(v)와의 공유키($K_{u,v}$)를 통해 메시지와 검증 메시지를 암호화 하여 전송한다. M은 메시지를 나타내며 V는 검증 메시지를 의미한다. 이때 V는 (수식 3)과 같이 구해진다.

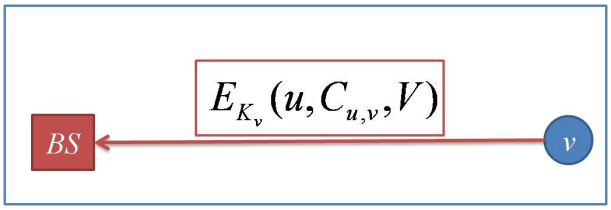
$$V = H(C_{u,v} \| K_u) \qquad (3)$$

(그림 2)는 검증값 전송 시의 데이터 전송을 나타낸다. 이는 수신 노드가 요청한 주기마다 수행하며, 이때 안전 노드 목록의 노드 중 $(1 + s \bmod S)$ 번째 노드로 (그림 2)의 <메시지 1>을 전송한다. 이때 S는 수신 노드에 인접한 안전 노드의 개수를 나타내며, s는 M의 순서 번호를 의미한다. 이와 같은 방법으로 안전 노드를 선택하는 이유는 안전 노드 역시 물리적 획득 공격에 취약하기 때문에 목록의 여러 안전 노드를 다양하게 선택하기 위함이다. <메시지 1>을 수신한 안전 노드 S_i는 <메시지 1>을 복호화 하여 해당 메시지의 목적지 v를 확인한 후 수신 노드와의 유일한 키로 암호화한 <메시지 2>를 수신 노드에게 전송한다.

안전 노드를 통해 안전하게 검증값을 전달 받은 수신 노드는 자신의 검증값과 송신 노드의 검증값을 비교한다. 만약 두 값이 동일하면 송신 노드와 수신 노드 둘간의 공유키는 안전한 상태라고 판단한다. 하지만 두 값이 다르면 오용키를 통한 공격이 있은 것으로 판단하고, BS에 공유키를 검증하기 위한 메시지를 만들어 전송한다. 이 메시지는 (그림 3)의 메시지와 같다.



(그림 2) 검증값 전송 시 데이터 전송



(그림 3) BS로 검증 메시지 전송

이때 수신 노드가 클러스터 헤더라면, BS에 보고해야 하는 다른 노드로부터 수신한 데이터가 존재한다. 이러한 데이터들은 에너지 효율을 위해 데이터 축약(data aggregation)[3]과정을 통해 하나의 보고 메시지로 합쳐진 후 BS로 전송된다. 따라서 BS에 검증 받을 공유키가 있다면 보고 메시지에 해당 공유키에 대한 검증 메시지를 붙여 하나의 메시지로 전송하면 메시지 전송 횟수를 줄여 클러스터 헤더의 자원을 효과적으로 절약할 수 있다.

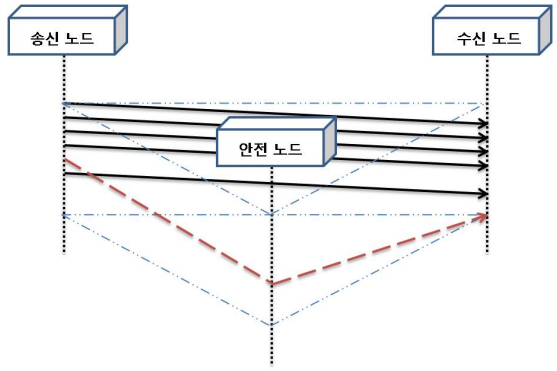
3.2 BS의 검증 과정

BS는 수신 노드로부터 받은 검증 메시지와 검증값을 통해 오용키에 대해 검증하고, 오용키를 발견하면

해당 오용키를 폐기한다. 그 과정은 다음과 같다. BS는 전송 받은 $C_{u,v}$ 에 센서 노드 u 의 비밀키(K_u)를 붙인 값을 해쉬한 값과 수신한 검증 메시지(V)를 비교하여 검증을 한다. 즉, 수신 노드의 $C_{u,v}$ 와 송신 노드의 단일키를 통해 검증 메시지 $V'=H(C_{u,v}|| K_u)$를 생성하고, V'와 V 를 비교하여 이 두 값이 다르다면 해당 공유키를 폐기한다.

3.3 두 노드 사이의 동기화를 고려한 DAC 기법

DAC 기법은 짧은 시간 동안 다수의 패킷을 전송할 경우, 두 노드의 검증값이 동기화가 이루어지지 않는 문제가 발생한다. (그림 4)가 이러한 시나리오를 나타낸다. 송신 노드는 자신이 전송하는 순서대로 검증값을 갱신하는데 비해, 수신 노드는 자신이 수신하는 순서에 따라 검증값을 갱신한다. 그런데 안전 노드를 통해 전달 되는 메시지는 직접 전달하는 메시지보다 전달되는데 오랜 시간이 소요되므로, 연속적으로 패킷을 전송할 경우 안전 노드를 통해 전송한 패킷보다 후에 전송한 패킷이 수신 노드에는 먼저 도착하여 수신 노드의 검증 값을 갱신할 수 있다. 이 경우 공격자에 의해 삽입된 메시지가 없음에도 불구하고 수신 노드와 송신 노드의 검증값이 다르게 되어 안전한 공유키를 폐기하는 문제점이 발생한다.



(그림 4) 검증값($C_{u,v}$) 동기화 문제

하지만 무선 센서 네트워크에서 위의 시나리오 같이 짧은 시간 동안 빈번하게 패킷을 전송하는 일은 매우 드물다. 보편적으로 무선 센서 네트워크에서 센서 노드들은 일정한 주기를 가지고 주변 환경을 감지하여 그 정보를 BS 에게 전송하거나, 특정한 이벤트가 발생하면 그 정보를 BS 에 알리며, 해당 데이터는 매우 작은 크기를 가진다.

기본 DAC 기법은 보편적인 계층적 라우팅을 사용하는 무선 센서 네트워크를 대상으로 제안된 기법이기 때문에 이와 같은 검증값의 동기화를 고려하지 않는다. 검증값의 동기화를 고려해야 하는 특수한 환경에서 본 기법을 사용하기 위해서는 약간의 수정이 필요하다.

동기화 문제의 원인은 안전 노드를 통해 전달되는 메시지와 그 이후에 전송한 메시지의 순서가 뒤바뀌어 수신 노드에 전달되어 수신 노드의 검증값이 송신 노드와 다른 순서로 갱신 되는 것이다. 따라서 이를 해결하기 위해서는 수신 노드가 올바른 순서로 검증값의 갱신해야 한다. 따라서 문제를 야기시키는 안전 노드를 통해 전달되는 메시지는 검증값 갱신에 사용하지 않는다. 이 메시지는 안전한 경로로 전달되므로 검증이 필요없다. 두 번째로 검증값이 갱신되더라도 검증 메시지보다 그 이후의 메시지가 수신 노드에 먼저 도착하면 수신 노드의 검증값이 갱신되어 동기화가 어긋난다. 이를 막기 위해 수신 노드는 임시 검증값을 사용한다. 이 값은 기존의 검증값과 동일하게 생성 갱신되며, 대신 기존 수신 노드의 검증값의 갱신 과정이 바뀌었다. 검증값은 수신 메시지마다 갱신되지 않고, 순서 번호가 $p \times n - 1$인 메시지를 수신할 경우 검증값을 임시 검증값으로 갱시한다. 여기서 p 는 검증 메시지를 요청한 주기가 되고, n 은 자연수이다. 이를 통해 검증 메시지 이후의 메시지가 먼저 수신되더라도 검증값은 동일한 값을 유지한다.

이와 같은 방법으로 두 노드 사이의 검증값의 동기화 문제를 해결할 수 있다. 하지만 이 방법을 사용하게 되면 수신 노드에 추가적으로 저장공간인 필요하고, 특히 수신 노드가 클러스터 헤더라면 클러스터내의 센서 노드의 수만큼 추가적인 자원이 소모된다.

4. 결론

본 논문에서는 오용키를 통해 무선 센서 네트워크 내에 수정된 메시지를 유입하는 행위를 검증하는 DAC 기법을 제안하였다. 이 기법은 특히 계층적 라우팅을 사용하는 무선 센서 네트워크 환경에서 보다 적합하다. DAC 기법은 기존의 분산 검증 기법과 달리 수신 노드가 추가적인 검증 메시지를 다른 검증자에게 전송하지 않고, 자신이 직접 검증값을 비교하기 때문에 매우 효과적이며, 이러한 특징은 클러스터 헤더에서 더욱 큰 효과를 발휘한다. 또한 클러스터 헤더는 데이터 축약과정 후 데이터를 BS 에게 전송하기 때문에, 만일 클러스터 헤더에서 노출된 공유키를 검증하게 되면 검증 메시지를 전송할 데이터에 붙여서 전송하면 되기 때문에 보다 효과적이다.

앞으로 제안한 DAC 기법의 성능평가 방안을 모색하고, 이를 바탕으로 보다 효율적인 오용키 검증 기법에 대한 연구를 진행할 것이다.

참고문헌

[1] Y. Wang, G. Attebury, and B. Ramamurthy, “A Survey of Security Issues in Wireless Sensor Networks”, IEEE Communications Surveys & Tutorials, vol. 8, no. 2, 2nd Quarter 2006

[2] D. Liu and Q. Dong, “Detecting Misused Keys in Wireless Sensor Networks”, IPCCC2007: Perform. Comp. and Comm. Conf., pp.272-280, April 2007

[3] E. Fasolo and M. Rossi, “In-Network Aggregation Techniques for Wireless Sensor Networks: A Survey”, IEEE wireless communication, Apr. 2007.

[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 Communication Security, pp. 41-47, Nov. 2002.