무선 센서 네트워크에서의 통신 근원지 및 도착지 은닉(제1부) : 프로토콜 설계

차영환*

Concealing Communication Source and Destination in Wireless Sensor Networks(Part I) : Protocol Design

Yeong-Hwan Tscha*

요 약

무선 센서 네트워크에 있어서 전역 도청에 대응하는 통신 근원지와 도착치의 위치기밀을 유지하기 위해서는 수많은 속임수용 더미 패킷들이 지속적으로 발행된다. 근원지나 도착지를 포함하는 특정 크기의 보호구역들을 설정한 후, 이 안의 노들들에 대해서만 데이터전송 중에 더미 패킷들을 생성하도록 제한하여 근원지와 도착지의 위치기밀성을 제공하면서도 패킷발생량을 줄이는 접근방안을 고려한다. 제안 프로토콜을 설계하고 형식화한 상세한 프로토콜 명세서를 제시한 후, 주요 특성과 정확성을 검증한다.

ABSTRACT

Against the global eavesdropping in wireless sensor networks, tremendous amount of dummy packets for faking are likely to be continuously generated in order to keep the location privacy of the communication source and destination. In our approach only certain disk-shaped zones of encompassing sources and destination are allowed to issue dummy packets during the data transfer so that the amount of generated packets is reduced while the location privacy of the source and destination remains secret. To this end we design a routing protocol and propose a detailed formal specification of it, and verify major characteristics.

키워드

Global eavesdropping, Location privacy of source and destination, protocol design and verification
광역도청, 근원지 및 도착지 위치기밀, 프로토콜 설계 및 검증

Ⅰ. 서 론

무선 센서 네트워크 라우팅 차원에서 통신 노드(node)의 위치기밀(location privacy)을 유지하기 위한 대부분의 연구는 도청자(eavesdropper)의 도청능력이 일반 통신 노드가 패킷을 전송할 때 발생하는 무선 신호를 감지하는 능력과 동일한 지역도청(local eavesdropping)을 고려한 것이다[1]. 즉, 데이터패킷의 전송에 따른 발생하는 신호를 따라 1-홉(hop)씩 거슬러 올라가면서 통신 근원지(source) 노드의 위치를 파악하려는 공격에 대응하는 근원지 위치보호 라우팅기법[2],[3]이나 전송패킷 신호를 따라 옮겨가며 도착지(destination) 노드의 위치를 발견하는 공격에 대응하는 도착지위치보호 라우팅기법[4] 등으로 분류된다.

* 상지대학교 컴퓨터정보공학부(yhtscha@sangji.ac.kr)
접수일자 : 2012. 12. 20 심사(수정)일자 : 2012. 12. 28 게재확정일자 : 2013. 02. 20

이러한 기법들은 공통적으로 길이가 긴 경로를 사용하면서 무작위로 다음-홉(hop)을 선정하여 추적자(즉, 도청자)를 따돌리기 위한 속임수용 더미(dummy) 패킷을 경로 주변 노드로 하여금 발행한다[5]. 또한 단일 경로가 아닌 병렬 경로 또는 분기가 발생하는 경로를 이용하기도 한다. 그러나 도청 노드들을 별도로 설치하여 도청 대상 네트워크 전체를 감시할 수 있는 광역도청(global eavesdropping)에서는 기존의 지역도청에 대응하여 위치기밀을 유지하기 위한 고려된 라우팅기법은 유용하지 못하다. 왜냐하면 일련의 데이터패킷들을 반복해서 발생해 내는 근원지나 이러한 패킷들을 최종적으로 수신하는 도착지가 고정되므로 광역도청에서는 쉽게 파악되기 때문이다.

광역도청에 의한 근원지와 도착지의 위치를 알아내려는 공격에 대응하는 라우팅기법으로는 가장 최근에 발표된 PCM(Periodic Collection Method)이 거의 유일하다[6]. PCM은 단순하면서도 높은 위치보안성을 제공하지만, 네트워크 전체의 모든 노드들이 매 시각마다 패킷을 발행하여야 하므로 과중한 통신비용을 유발하여 대규모 네트워크에서는 실용적인 라우팅방법이 되지 못한다.

제안하는 라우팅기법 ELPR(End-node Location Privacy Routing)은 통신 경로 상의 근원지와 도착지의 위치기밀을 동시에 고려하는 프로토콜이다. 데이터패킷 전송 동안에 근원지와 도착지를 포함하는 위치보호구역(location privacy zone)이라는 원(disk) 모양의 일정 범위내의 노드들과 보호구역 밖의 경로를 구성하는 노드들만이 타임-슬롯마다 데이터패킷 또는 더미패킷을 발행하도록 함으로써 전체적인 패킷 발생비용을 줄이는 한편, 도청자로 하여금 보호구역내의 어느 노드가 근원지인지 또는 도착지인지 다른 노드와 구별될 수 없게 한다. 보호구역의 크기는 보안성을 고려하여 확장 축소할 수 있으며 이에 따라 통신비용이 비례하는 장점이 있다.

II. 가정 사항

공격자는 네트워크 전체에 걸친 도청을 위한 별도의 감시 네트워크를 구축하고 있어 임의의 노드의 패킷 전송에 의한 신호 발생을 즉시 감지할 수 있다고 가정한다. 그리고 노드들 간의 통신에 어떠한 방해도 주지 않으며, 노드 내부로의 침투나 기능 변경 또는 물리적 파괴 등의 공격은 수행하지 않는 수동적 광역 도청에 의해 패킷이 생성되는 근원지나 최종 수신자인 도착지를 찾아내는 것을 목적으로 한다.

센서 노드들은 시간적으로 동기화 되어 이포크(epoch)라는 기간 단위로 데이터의 수집이 반복 진행된다고 가정한다. 이를 위해 모든 노드는 이포크 기간을 나타내는 타이머(timer) T[epoch]를 유지하며, 하나의 이포크는 여러 개의 타임 슬롯으로 구성된다. 패킷은 매 타임 슬롯에 맞추어 송수신된다.

논의의 편의를 위해, 고려되는 패킷 전송에서는 어떠한 오류도 발생하지 않으며 모든 패킷은 적절한 암호화기법에 의해 전송되는 내용의 기밀이 유지된다고 가정한다. 근원지는 전송할 데이터가 있는 임의의 노드를 말하고, 근원지가 발행한 패킷은 중간의 중계 노드들을 거쳐 최종적으로 도착지에 이른다. 여기서, 도착지는 네트워크 내의 기지국(basestation)이다.

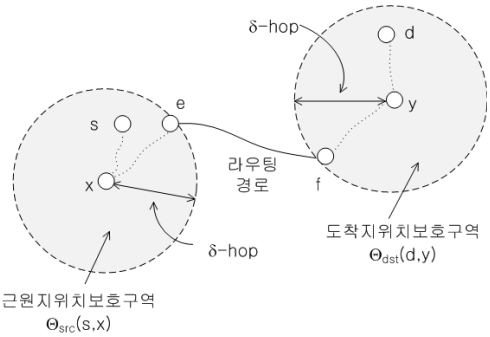

그림 1. ELPR의 접근방법
Fig. 1 The ELPR approach

III. 접근방법

제안하는 라우팅 프로토콜 ELPR의 접근방법은 그림 1과 같이 설명할 수 있다. 데이터를 전송할 근원지나 이를 수신할 도착지가 직접 보호구역을 설정하면 구역의 중앙에는 바로 근원지나 도착지가 위치하기 때문에 위치가 바로 드러난다. 고로 근원지나 도착지는 자신을 대신해서 보호구역을 설정하고 데이터패킷의 송신이나 수신을 대신해 줄 프락시(proxy)를 선정

한다. 이포크가 시작되면 네트워크 내의 모든 노드는 자신으로부터 δ-홉(hop) 이내의 노드들에게 자신의 존재를 알리는 패킷을 보낸다. 이에 데이터를 전송할 근원지나 수신할 도착지는 자신으로부터 δ-홉(hop) 이내에 존재하는 노드들 중에서 일정 요건을 만족하는 노드를 선정하여 자신의 프락시로 삼는다.

그림 1은 근원지 s와 도착지 d가 각기 자신으로부터 δ-홉(hop) 이내에 존재하는 노드들 중 x와 y를 각기 근원지 프락시와 도착지 프락시로 정하고 이들을 통해 δ-홉(hop) 이내의 노드들로 구성되는 원 모양의 근원지보호구역 $\Theta_{src}(s,x)$와 도착지보호구역 $\Theta_{dst}(d,y)$을 설정한 상황이다. 보호구역내의 모든 노드는 매 타임-슬롯마다 패킷을 발행하므로 도청자는 어느 노드가 근원지이고 도착지인지 선별할 수 없다. 보안수준에 의거 δ 값을 증감하는 정도에 따라 보호구역내의 노드들의 수가 정해지게 되므로 위치보안과 통신비용 간의 절충이 가능하다. 근원지프락시 x와 도착지프락시 y간에는 데이터패킷을 전송하기 위한 라우팅경로가 설정되는데 영역경계에 있는 두 노드 e와 f사이에는 단일경로를 사용한다. 그림에서는 근원지가 하나인 경우를 보여주고 있지만 근원지는 여러 개일 수 있다. 다만, 도착지는 언제나 기지국 하나임을 가정한다.

Ⅳ. 프로토콜 설계

4.1 동작단계 및 상태

ELPR은 크게 두 가지 상태 즉, 초기화단계와 데이터전송단계를 거치며 사건-기반(event-driven) 형태로 수행된다. 사건은 다른 노드로부터 패킷이 도착하거나 사용 타이머 기간이 종료되는 것과 같이 일정 요건을 만족하는 상황을 의미한다. 보호구역 밖의 노드들의 경우 데이터패킷의 전송을 위한 경로상의 노드들 제외한 모든 노드들은 전원절약을 위한 휴면(sleeping) 상태로 다음 이포크까지 남는다. 보호구역 밖의 노드들의 경우 데이터패킷의 전송을 위한 경로상의 노드들 제외한 모든 노드들은 전원절약을 위한 휴면(sleeping) 상태로 다음 이포크까지 남는다.

그림 2에 ELPR의 상태천이 관계를 나타내었다. 이포크 시작과 함께 각 노드는 자신의 존재를 자신으로부터 δ-홉 이내의 노드들에게 알리는 과정이 수행하면서 초기화단계(즉, "INITIAL_PH")로 접어든다. 이 단계에서 각각의 근원지와 도착지 모두는 자신의 프락시를 정한다. 그리고 프락시들을 통해 자신들이 설정한 보호구역을 브로드캐스트 함으로써 모든 노드에게 알린다. 그러면 이를 수신한 노드들은 어느 노드(즉, 프락시)를 중심으로 δ-홉 이내의 원 형태의 보호구역이 설정되었는지 그리고 그 프락시로부터 자신이 얼마만큼 떨어져 있는지 알게 된다. 데이터패킷의 도착지인 기지국으로부터 데이터 전송이 가능함을 알리는 패킷이 도착하면 데이터전송단계(즉, "DATA_PH")로 접어드는데, 각 보호구역의 프락시와 자신과의 거리를 알므로 어떠한 보호구역 안에도 속하지 않는 노드는 초기화단계에서 바로 휴면상태(즉, "SLEEPING")가 된다. 다만, 휴면상태의 노드 중에 데이터전송단계에서 발생하는 데이터패킷을 전달하여야 하는 라우팅 경로상의 노드는 데이터전송 상태로 전환한다.

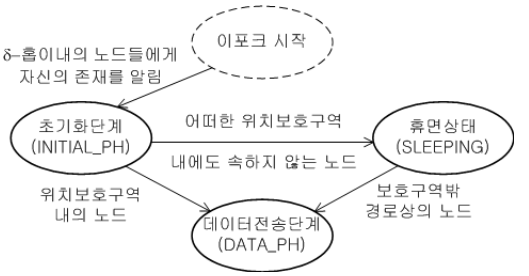

그림 2. 프로토콜 상태
Fig. 2 Protocol state

4.2 패킷형식과 구성필드

ELPR에서의 패킷 유형(type)와 필드(field)는 표 1과 같다. HL(Hello), FP(Flood PZ), PZ(Privacy Zone)는 초기화단계에서 위치보호구역을 설정하는데 사용되는 패킷들이며, FR과 RR은 도착지 보호구역이 결정되고 도착지가 데이터를 수신할 수 있음을 알려 데이터전송단계로 접어들게 한다. 근원지는 자신의 프락시에게 전송데이터를 담은 FD를 보내고, 프락시는 이를 이용하여 DT를 도착지보호구역으로 전송함으로써 데이터가 전달된다. F로 시작하는 패킷들 즉, FP, FR, FD는 근원지 또는 도착지가 자신의 프락시에게 보내는 패킷들로 그 수신 노드들은 δ-홉 이내에 있는 노드들로 제한된다. PZ와 RR은 근원지 프락시와 도착지 프락시에 의해 모든 노드들로 전달된다.

표 1. 패킷 종류와 필드
Table 1. Packet types and fields

필드 / 유형	Origin	Sink	Demander	Next_Hop	Hop_Count	User_Data
HL(hello)	√				√	
FP(flood PZ)	√	√			√	
PZ(Privacy Zone	√		√		√	
FR(flood RR)	√	√			√	
RR(receive ready)	√		√		√	
FD(flood DT)	√	√			√	√
DT(data)	√	√	√	√		√

표 2. 노드에서의 주요 변수
Table 2. Major variables at each node

변수	설명
T[Epoch]	이포크 기간을 나타내는 타이머
T[Proxy]	근원지프락시를 등록하는 기간을 나타내며 도착지에서 구동되는 타이머
δ	원 형태의 위치보호구역 반경에 해당하는 홉 수
MyAddress	노드 자신의 주소
Destination	도착지 즉, 기지국 주소
State	노드의 상태("INITIAL_PH", "DATA_PH, "SLEEPING")
SrcProxy[s]	근원지 s의 프락시
HopCountTo[j]	노드 j로의 최단경로길이
DestProxy	도착지 프락시
NextHopToDestProxy	도착지 프락시로 전달을 위한 다음-홉 노드

구성필드에서 Origin은 해당 패킷을 생성한 노드를, Sink는 해당 패킷을 수신할 노드이다. Demander는 해당 패킷을 발행하도록 요청한 노드를 나타내며 PZ의 경우에는 근원지이며, RR의 경우에는 도착지이다. Next_Hop은 데이터패킷 DT를 전달시 다음에 수신하여야 할 노드를 지정한다. Hop_Count는 패킷이 Origin으로부터 발생되어 이웃 노드를 거쳐 갈 때마다 1씩 증가한다. 이에 의해 패킷을 수신한 노드로 하여금 Origin으로부터 얼마만큼 떨어져 있는지를 알 수 있다. 데이터패킷 내의 사용자 정보는 User_Data 필드로 나타낸다. 패킷 필드 중, 패킷의 유형을 구분하는 필드나 발행 패킷의 선후관계와 중복 여부를 알기 위해 사용되는 순서번호 필드와 통상적 필드들은 나타내지 않았다. 또한, 패킷유형별로 구성필드는 다를지라도 전송 전에 모든 패킷의 길이가 동일하도록 조정되어 전송됨을 가정한다.

각 노드가 유지하는 주요 변수들은 표 2와 같다. 사용 타이머로는 이포크 기간을 나타내는 T[Epoch]외에 T[Proxy]를 두어 각각의 근원지로 하여금 자신들의 프락시를 설정하고 네트워크 내의 노드들에게 알리는 기간을 정한다. T[Proxy]가 만료면 기지국은 자신의 프락시를 통해 데이터의 수신이 가능함을 통보하고, 이어서 근원지로부터 데이터전송이 시작되게 된다. 표 2의 변수 중, SrcProxy[s]는 근원지 s의 프락시를 기억하는 변수로써 FA를 수신할 때마다 해당 근원지의 프락시를 저장하는 변수이다. DstProxy는 도착지 프락시를 저장하며, NextHopToDstProxy는 도착지 프락시와의 거리(즉, 홉 수)를 기억하는 변수이다.

4.3 동작 절차

1) 초기화 단계

초기화단계의 기능을 수행하는 명세서는 그림 3과 같다. 이포크의 시작과 함께 모든 노드가 case 1을 수행하게 되어 HL을 발행한다. case 2에서는 $0<i_1<i_2<\delta$ 인 관계가 성립하는 임의의 두 정수 i_1와 i_2를 선정하여 $i_1<$HL.Hop_Count$<i_2$를 만족하는 HL이 가장 먼저 도착하면(first-fit 규칙 적용) 이를 보낸 노드들 자신의 프락시로 정한다.

case 3과 case 4는 FA와 PZ가 각기 근원지와 근원지 프락시에서 발행되는 과정이다. case 5는 PZ를 수신한 노드 자신이 어느 보호구역에 속하는지를 판별하기 위해 HopCountTo[PZ.Origin] ← PZ.Hop_Count를 수행하여 자신과 PZ.Origin 즉, 근원지 프락시와의 거리를 얻는다. case 6의 역할은 더미패킷을 발행하는 것이다. 패킷 RR을 수신하여 초기화단계가 종료될 때까지 새로이 생성하거나 중계할 패킷이 없다면 case 1에서 발행하였던 HL을 다시금 발행하여 항상 패킷을 전송중인 것처럼 보이도록 한다.

도착지 프락시로 전달하기 위한 다음-홉 노드는 RR을 처음으로 정해준 노드가 된다. 왜냐하면 RR은 브로드캐스트가 되기 때문에 자신에게 가장 먼저 RR을 전해준 이웃 노드가 바로 RR을 발행한 도착지프락시로 가는 최단경로상에 존재하기 때문이다. 또한, RR.Hop_Count$\leq\delta$이라면 도착지보호구역에 자신이 속함을 알 수 있다.

```
// *** 초기화단계: 근원지 위치보호구역 설정
// 새로운 이포크(epoch) 시작
case 1: 타이머 T[Epoch]가 시작되어 새 이포크 시작
  1.1 initialize variablesand State←"INITIAL_PH";
  1.2 if this is basestation, re-start T[Proxy];
  1.3 create packet HL such that
      HL.Origin←MyAddress, HL.Hop_Count←0;
  1.4 send HL and copy and name it RetxHL;

// 자신의 프락시를 결정
case 2: 중복되지 않은 HL이 수신되고
HL.Hop_Count≤δ임
  2.1 if (MyProxy==∅ and either source or
destination) {
      choose integers i1 and i2 such that  0<i1<i2<δ;
      if (i1<HL.Hop_Count<i2) { // first-fit 규칙
        if (this is source) // 근원지프락시
          SrcProxy[MyAddress]←HL.Origin;
        else // 도착지프락시
          DstProxy←HL.Origin; }
  2.2 if (HL.Hop_Count<δ) { // HL전파
      HL.Hop_Count←HL.Hop_Count+1; send HL; }

// 근원지는 프락시에게 보호구역을 알릴것을 요청
case 3: MyProxy≠∅이면서 전송할 데이터가 있음
  3.1 create packet FP such that
FP.Origin←MyAddress,
      FP.Sink←SrcProxy[MyAddress],
FP.Hop_Count←0;
  3.2 send FP;

// 근원지프락시는 네트워크 전체로 보호구역을 알림
case 4: 중복되지 않은 FP가 수신되고
FP.Hop_Count≤δ 임
  4.1 if (MyAddress==FP.Sink) { // 근원지 프락시
      make packet PZ such that
PZ.Origin←MyAddress,
        PZ.Demander←FP.Origin, PZ.HopCount←0;
      send PZ; }
  4.2 if (FP.Hop_Count<δ) { // PZ를 전파
      FP.HopCount←FP.HopCount+1; send FP; }

// 근원지, 근원지 프락시 및 프락시와의 거리를 저장
case 5: 중복되지 않는 PZ가 도착함
  5.1 SourceProxy[PZ.Origin]←PZ.Demander; // 근원지
  5.2 HopCountTo[PZ.Origin]←PZ.Hop_Count; // 거리
  5.3 PZ.Hop_Count←PZ.Hop_Count+1 and send PZ;

// 빈(empty) 타임-슬롯에서 더미패킷을 발행
case 6: State=="INITIAL_PH"이며 수신 패킷없음
  6.1 send RetranHL;
```

그림 3. 초기화 단계 절차
Fig. 3 Procedure of the initialization phase

2) 데이터전송 단계

그림 4의 case 7은 T[Proxy]가 종료되어 기지국에서 데이터전송단계로의 전환 즉, State←"DATA_PH"를 수행하고, 데이터를 수신할 준비가 되어 있음을 FR을 이용하여 자신의 프락시에게 알리는 과정이다. case 8에서 FR을 수신한 도착지 프락시는 도착지위치보호구역을 알리고자 RR을 브로드캐스트 한다. case 9의 9.1~9.3은 각기 RR을 수신한 노드가 도착지, 도착지 프락시 그리고 도착지 프락시로 데이터패킷을 전송할 때 사용할 다음-홉 노드를 저장한다.

도착지 프락시로 전달하기 위한 다음-홉 노드는 RR을 처음으로 정해준 노드가 된다. 왜냐하면 RR은 브로드캐스트 되기 때문에 자신에게 가장먼저 RR을 전해준 이웃 노드가 바로 RR을 발행한 도착지프락시로 가는 최단경로상의 노드가 되기 때문이다. 도착지 프락시까지의 거리는 9.4에서 결정 되므로, 9.5에서는 현재까지 RR을 통해서나 초기화단계에서의 PZ를 통해 알게 된 보호구역에 자신이 속하는지 판별한다. 도착한 RR에 대해 RR.Hop_Count≤δ이라면 도착지보호구역에 속하는 것이다. 그리고 어떤 근원지 k에 대해 HopCountTo[SrcProxy[k]≤δ인 경우에는 k의 보호구역에 자신이 속하는 것이므로 프로토콜 상태는 데이터전송단계로 전환된다. 그 밖에 어떠한 보호구역에도 속하지 않는 노드는 9.7에서 휴면상태가 된다. case 10과 case 11에서는 근원지가 데이터 전송을 위해 자신의 프락시에게 FD를, 프락시는 이에 대응하는 DT를 발행하는 과정이다.

case 12에서는 State=="DATA_PH"인 즉, 데이터전송단계 상태에 있는 노드만 데이터패킷 DT를 전송한다. case 12에서 DT를 수신한 노드는 자신이 경로상의 다음-홉 노드가 아니더라도 DT를 재 전송하는데, 이는 DT를 더미패킷으로 활용하는 것이다. 근원지 보호구역과 도착지 보호구역이 겹쳐져 있지 않는 경우에는 휴면상태에 있는 노드들이 DT를 중계해주어야 한다. case 13은 이를 위한 것으로 휴면상태에 있던 노드들 중 DT.Next_Hop에 명기된 노드는 데이터전송단계의 상태로 바뀌면서 다음-홉 노드로 DT를 전달한다. 이후 발생하는 DT의 경우는 case 12의 12.3을 수행하게 된다. case 14는 데이터전송단계에 있는 노드가 빈(idle) 타임-슬롯에서 더미패킷을 발행하는 경우이다. 프로토콜 상태가 "SLEEPING"인 즉,

휴면상태에 있는 노드들은 더미패킷을 발행하지 않게 되어 기존의 PCM[6]에 비해 통신비용의 감축을 가져온다.

```
// *** 데이터전송단계: 도착지보호구역 설정과 데이터 전송
// 프락시로 하여금 데이터 수신이 가능함을 알리도록 함
case 7: 기지국에서 T[Proxy]가 종료되고 MyProxy ≠∅임
  7.1 if (State=="INITIAL_PH") {
        State←"DATA_PH"; // 데이터전송상태로 이동
        create packet FR such that
FR.Origin←MyAddress,
          FR.Sink←DstProxy, FR.Hop_Count←0;
        send FR; }

// 도착지 프락시는 RR을 만들어 네트워크 전체로 전송
case 8: 중복되지 않은 FR이 수신되고,
FR.Hop_Count≤δ임
  8.1 if (FR.Sink==MyAddress) { // 도착지프락시일 경우
      State←"DATA_PH"; Destination←FR.Origin;
      DstProxy←MyAddress;
      create packet RR such that
RR.Origin←MyAddress,
        RR.Demander←FR.Origin, RR.Hop_Count←0;
      send RR; }
  8.2 if (FR.Hop_Count<δ) // FR을 전파
      FR.Hop_Count←FR.Hop_Count+1 and send FR;

// 도착지, 도착지 프락시 및 프락시와의 거리 저장
case 9: 중복되지 않은 RR이 이웃노드 u로부터 도착함
  9.1 Destination←RR.Demander; // 도착지(기지국)
  9.2 DstProxy←RR.Origin; // 도착지프락시
  9.3 NextHopToDstProxy←u; // 다음-홉
  9.4 HopCountTo[RR.Origin]←RR.Hop_Count; // 거리
  9.5 if (RR.Hop_Count≤δ) // 도착지보호구역내 존재
      State←"DATA_PH";
    else if (HopCountTo[SrcProxy[k]]≤δ for some k)
      State←"DATA_PH"; // 근원지보호구역내 존재
  9.6 RR.Hop_Count←RR.Hop_Count+1 and flood RR;
  9.7 if (State≠"DATA_PH") // 보호구역 밖에 위치함
      State←"SLEEPING"; // 휴면상태

// 근원지는 프락시를 통해 데이터패킷을 전송케 함
case 10: State=="DATA_PH"인 근원지 노드
  10.1 create packet FD such that
FD.Origin←MyAddress,
        FD.Sink←SrcProxy[MyAddress],
        FD.Hop_Count←0, FD.User_Data←UserData;
  10.2 send FD;

// 근원지 프락시는 DT를 전송함
case 11: FD is received where, FD.Hop_Count≤δ
  11.1 if (FD.Sink==MyAddress) { // 도착지
       create packet DT such that
         DT.Origin←MyAddress,
DT.Sink←Destination,
         DT.Demandert←FD.Origin,
         DT.Next_Hop←NextHopToDstProxy,
         DT.User_Data←UserData;
       flood DT; }
    else if (FD.Hop_Count<δ) { // FD를 전파
         FD.Hop_Count←FD.Hop_Count+1; send FD; }
```

그림 3. 초기화 단계 절차
Fig. 3 Procedure of the initialization phase

```
// DT를 수신한 보호구역내의 노드
case 12: State=="DATA_PH"이고 중복되지 않은 DT를 수신
  12.1 if (HopCountTo[DstProxy]≤δ) { //
         도착지보호구역
       if (DT.Sink==MyAddress) // 도착지에 도착
         accept DT.User_Data;
       else if (DT.Next_Hop==MyAddress)
         DT.Next_Hop←NextHopToDstProxy; //
         다음-홉
         ;
       send DT; }
  12.2 if (HopCountTo[DT.Origin]≤δ) { //
         근원지보호구역
       if (DT.Next_Hop==MyAddress)
         DT.Next_Hop←NextHopToDstProxy; //
         다음-홉
         ;
       send DT; }
  12.3 if (DT.Next_Hop==MyAddress) { // case 13 후속
       DT.Next_Hop←NextHopToDstProxy;
       send DT; }

// 보호구역들간의 경로 상에 존재하는 노드
case 13: State=="SLEEPING"이며
  MyAddress==DT.Next_Hop인 DT 도착
  13.1 State←"DATA_PH"; // 데이터전송단계로 전환
  13.2 DT.Next_Hop←NextHopToDstProxy; // 다음-홉
  13.3 send DT;

// 보호구역내 노드로 빈 타임-슬롯에서 더미 패킷발행
case 14: State=="DATA_PH"이며 수신패킷 없음
  14.1 send RetxHL previously copied in case 1;
```

그림 4. 데이터전송 단계 절차
Fig. 4 Procedure of the data transfer phase

Ⅴ. 검 증

근원지 s의 프락시 x로부터 δ-홉에 이내에 존재하는 모든 노드들을 포함하는 보호구역을 $\Theta_{src}(s,x)$로, 마찬가지로 d와 그의 프락시 y에 대해 성립하는 도착지 보호구역 $\Theta_{dst}(d,y)$를 고려하자(그림 1 참조). 네트워크는 단절되어 있지 않아 어떠한 노드도 다른 노드들과 통신경로를 설정할 수 있다고 가정한다.

성질 1. $\Theta_{src}(s,x)$에는 근원지 s가, $\Theta_{dst}(d,y)$에는 도착지 d가 반드시 포함된다.

설명: 그림 3의 case 2에서 근원지 s나 도착지 d는 임의의 두 정수 i_1와 i_2에 대해 $i_1<HL.Hop\_Count<i_2<\delta$ 인 관계를 만족하는 즉, 자신으로부터 i_1-홉보다는 멀리 그러나, i_2-홉을 넘지 않는 거리의 노드를 프락시로 택한다. 고로 $i_2<\delta$이므로 성질 1은 성립한다.

성질 2. $v\in\Theta_{src}(s,x)$인 임의의 노드 v와 $u\in\Theta_{dst}(d,y)$인 임의의 노드 u는 타임-슬롯마다 패킷을 발행한다.

설명 : 먼저 초기화단계를 고려한다. 그림 3의 case 1에 의해 모든 노드는 HL을 발행하고 이를 RetxHL이란 이름으로 복사하여 놓는다. 데이터를 전송할 근원지는 case 3에서 FP를 발행한다. 그리고 근원지프락시는 PZ를 발행한다. 이 후에는 case 6과 같이 HL, FP 및 PZ 등 어느 것도 수신하지 않은 경우에는 case 1에서 HL을 발행하며 복사해놓았던 RetxHL을 더미로 전송한다. 데이터전송단계에서는 그림 4의 case 7에서 도착지 d는 FR을, 도착지 프락시는 RR을 발행한다. 그리고 상태가 "DATA_PH"로 된다. 또한 case 9의 9.5에 의해 보호구역내에 속하는 다른 노드 즉 $v\in\Theta_{src}(s,x)$인 임의의 노드 v와 $u\in\Theta_{dst}(d,y)$인 임의의 노드 u 역시 상태가 "DATA_PH"로 된다. 근원지 s는 case 10에서 FD을, 프락시 x는 DT를 생성한다. 이후 데이터전송단계에서 발행한 이러한 패킷들을 다시 생성하지 않거나 또는 이러한 패킷들을 이웃 노드로부터 수신하지 않은 경우에는 case 14에 기술된 것처럼 상태가 "DATA_PH"인 노드들은 다시금 더미 패킷 RetxHL을 발행한다. 이에 보호구역내의 모든 노드는 매 타임-슬롯마다 패킷을 발행한다.

성질 3. $v\notin\Theta_{src}(s,x)$이고 $v\notin\Theta_{dst}(d,y)$이며 $\Theta_{src}(s,x)\rightarrow\Theta_{dst}(d,y)$인 경로 상에도 존재하지 않는 노드는 데이터전송 단계에서 휴면 상태에 머문다.

설명 : 성질 2를 만족하지 않는 노드는 어떠한 보호구역내에도 들어있지 않는 노드이다. 다만 근원지보호구역에서 도착지보호구역으로 데이터패킷 DT를 전송하기 위해 사용하는 라우팅경로 $\Theta_{src}(s,x)\rightarrow\Theta_{dst}(d,y)$ 중 보호구역 밖의 경로상의 노드들은 case 13에 의해 상태 값이 "DATA_PH"로 바뀌어 전송할 DT가 없는 경우에는 case 14에 의해 더미패킷을 발행한다. 따라서 성질 3이 성립한다.

성질 4. $\Theta_{src}(s,x)$의 x에서 $\Theta_{dst}(d,y)$의 y로 데이터패킷(DT)을 전달하는 경로 $\Theta_{src}(s,x)\rightarrow\Theta_{dst}(d,y)$는 x와 y 간의 최단경로이다.

설명: 그림 4의 case 9을 수행하는 근원지 s의 프락시 x가 도착지 y가 발행한 RR을 최초로 이웃노드 u로부터 받았다는 것은 y와 x사이의 최단경로는 u을 경유하는 것임이 틀림없다. 따라서 $\Theta_{src}(s,x)\rightarrow\Theta_{dst}(d,y)$는 x와 y간의 최단경로이다.

성질 5. ELPR에 의해 근원지 및 도착지의 위치기밀은 유지된다.

설명 : 성질 1, 2, 3에 의해 ELPR에서는 위치보호구역내의 노드들과 이들을 연결하는 즉, 위치보호구역을 설정한 프락시들을 이어주는 경로상의 노드들만이 데이터전송단계에서 매 타임-슬롯마다 패킷을 발행한다. 따라서 PCM[6]에 비해 불필요한 더미패킷의 발행을 억제하면서, 패킷전송 시 발생하는 신호를 감지하는 도청이 네트워크 전체에 걸쳐 수행되더라도 보호구역내의 어느 노드가 근원지 또는 도착지인지 다른 노드들과 구별할 수 없어 그 위치기밀이 유지된다.

Ⅵ. 결 론

이 연구에서는 무선 센서 네트워크에 있어서 전역도청에 대응하여 통신 경로의 근원지와 도착치의 위치기밀을 유지하기 위해 발행되는 속임수용 더미 패킷들을 줄이기 위한 라우팅 프로토콜 ELPR을 제안하였다. 근원지나 도착지를 포함하는 특정 크기의 보호구역들을 설정한 후, 이 안의 노들들에 대해서만 데이터전송 중에 더미 패킷들을 생성하도록 제한하여 근

원지와 도착지의 위치기밀성을 제공하면서도 패킷발생량을 줄이도록 하였다. ELPR을 설계하여 상세한 프로토콜 명세서를 제시하고 논리적 차원에서의 정확성을 검증하였다.

후속 연구에서는 ELPR이 제공하는 위치기밀성을 정보이론의 엔트로피 기준으로 평가하고, 시뮬레이션을 통한 통신비용을 관련연구[6]과 비교할 것이다. 한편, [7], [8]과 같이 전송정보의 무결성 제공을 위한 라우팅이나 보안 프레임워크 설계에 있어서 위치보안을 연계한 연구도 흥미로운 주제로 판단된다. 아울러, 전략적 특수 통신[9], 스마트폰 사용자[10], 클라이언트 컴퓨팅 기반 등에서의 위치보호에 대한 연구와 효과적 보안키 생성[12]에 의핸 내용보안을 겸하는 연구도 필요하다.

감사의 글

본 논문은 2011년도 상지대학교 교내 연구비 지원에 의한 것임을 밝힙니다.

참고 문헌

[1] N. Li, N. Zhang, S.-K. Das, and B. Thuraisingham, "Privacy preservation in wireless sensor networks : a state-of-the-art survey", Ad Hoc networks, Vol. 7, pp. 1501-1514, 2009.

[2] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, “Enhancing source-location privacy in sensor networking routing", Proc. of ICDCS’05, pp. 1-10, 2005.

[3] Y. Quyang, Z. Le, D. Liu, J. Ford, and F. Makedon, "Source-location privacy laptop-class attacks in sensor networks", Proc. of SecureComm’08, Paper No. 44, 2008.

[4] Y. Jian, S. Chen, Z. Zhang, and L. Zhang, "Protecting receiver-location privacy in wireless sensor networks", Proc. of INFOCOM’07, pp. 1955-1963, 2007.

[5] H. Chen and W. Lou, "From nowhere to somewhere: protecting end-to-end privacy in wireless sensor networks", Proc. of IPCCC’10, pp. 1-8, 2010.

[6] K. Mehta, D. Liu, and M. Wright, "Protecting location privacy in sensor networks against a global eavesdropper", IEEE Transaction on Mobile Computing, Vol. 11, No. 2, pp. 320-336, 2012.

[7] 이철승, “MANET 기반의 MD5 보안 라우팅에 관한 연구”, 한국전자통신학회논문지, 7권, 4호, pp. 797-804, 2012.

[8] 서우석, 박재표, 전문석, “네트워크 보안 인프라 구성을 위한 표준화된 플랫폼 디자인 방법론에 관한 연구”, 한국전자통신학회논문지, 7권, 1호, pp. 203-211, 2012.

[9] 정상래, 신현식, “NCW 및 전술데이터링크 기술개발 현황분석”, 한국전자통신학회논문지, 7권, 5호, pp. 991-998, 2012.

[10] 정상래, 신현식, “스마트폰의 사용자 경험이 만족에 미치는 영향 연구”, 한국전자통신학회논문지, 7권, 5호, pp. 1087-1093, 2012.

[11] 차병래, 김대규, 김남호, 최세일, 김종원, “클라우드 컴퓨팅 기반 스트리밍 미디어의 검색 가능 이미지 암호 시스템 설계”, 한국전자통신학회논문지, 7권, 4호, pp. 811-819, 2012.

[12] 정우열, 이선근, “근접 통신망의 보안성 향상을 위한 자기키 생성 알고리즘에 관한 연구”, 한국전자통신학회논문지, 7권, 5호, pp. 1027-1032, 2012.

저자 소개

차영환(Yeong-Hwan Tscha)

1983년 인하대학교 전자계산학과 졸업(이학사)

1985년 한국과학기술원 전산학과 졸업(공학석사)

1993년 인하대학교 대학원 전자계산학과졸업(이학박사)

1985년~1990년 한국전자통신연구원 선임연구원

1986년 NIST(미국) 방문과학자

2004년, 2011년 Boğaziçi 대학교(터키) 방문교수

1994년~현재 상지대학교 컴퓨터정보공학부 교수

※ 관심분야 : 네트워크 구조, 통신 프로토콜, 네트워크 보안