情報保護學會論文誌
第19卷 第6號, 2009. 12

Onion 기법을 사용하지 않는 효율적인 MANET 익명 라우팅 프로토콜*

이 승 윤,[1†] 오 희 국,[1] 김 상 진[2‡]
[1]한양대학교, [2]한국기술교육대학교

An Efficient Anonymous Routing Protocol Without Using Onion Technique in MANET*

Sung-yun Lee,[1†] Hee-kuck Oh,[1] Sang-jin Kim[2‡]
[1]Hanyang University, [2]Korea University of Technology and Education

요 약

트랩도어, onion, 익명 인증 등의 기법을 이용하여 MANET(Mobile Ad hoc Network)에서의 프라이버시를 보호하기 위한 많은 연구가 진행되고 있다. MANET에서의 프라이버시 보호는 세부적으로 ID 프라이버시, 위치 프라이버시, 경로 프라이버시에 대한 보호와, 세션간 메시지 비연결성을 만족시키는 것으로 나눌 수 있다. 기 제안된 방법들은 위치 프라이버시나, 경로 프라이버시의 보호가 미비한 경우가 많고 프라이버시 요구조건을 만족하기 위해 소요되는 암호화 연산비용이 비교적 크다. 본 논문에서는 위에서 제시한 프라이버시 요구조건을 만족하면서도 연산비용을 낮춘 더욱 효율적인 익명 라우팅 프로토콜을 제안한다. 제안된 기법은 onion이나 익명인증 등의 기법을 사용하지 않고 노드의 프라이버시를 보호하여 보다 효율적인 라우팅경로 설정이 가능하며, 경로 설정 과정에 관계된 모든 노드를 고려하여 연산량을 비교함으로써 보다 정확한 효율성 분석을 제공한다.

ABSTRACT

There have been a lot of researches on providing privacy in MANET (Mobile Ad hoc NETwork) using trapdoor, onion, and anonymous authentication. Privacy protection in MANET can be divided into satisfying ID privacy, location privacy, route privacy, and unlinkability between sessions. Most of the previous works, however, were unsatisfactory with respect to location privacy or route privacy. Moreover, in previous schemes, cryptographic operation cost needed to meet the privacy requirements was relatively high. In this paper, we propose a new efficient anonymous routing protocol that satisfies all the privacy requirements and reduces operation costs. The proposed scheme does not use onion or anonymous authentication techniques in providing privacy. We also provide a more accurate analysis of our scheme's efficiency by considering all the nodes involved in the route establishment.

Keywords: MANET, anonymous routing, privacy

접수일(2009년 6월 16일), 수정일(2009년 9월 25일), 게재확정일(2009년 10월 22일)

* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음. (NIPA-2009- C1090-0902-0035)

* 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임. (No. 2009-0080351)

† 주저자, sylee@igloosec.com

‡ 교신저자, sangjin@kut.ac.kr

I. 서 론

MANET은 통신 기반구조 없이 이동성을 가진 노드들에 의해 자율적으로 구성되는 네트워크를 의미한다. MANET은 노드의 이동성으로 인해 네트워크의 구조가 지속적으로 변화하고 그에 따라 라우팅 정보도 함께 변화하는 특성이 있다. 그러므로 MANET에서는 리엑티브(reactive) 방식의 라우팅에 대한 연구가 많이 진행되었다[1-3]. 최근에는 네트워크 통신의 프라이버시에 대한 관심이 높아 감에 따라 MANET을 이용함에 있어서도 프라이버시 보호에 대한 요구가 나타나게 되었다. MANET에서의 프라이버시 보호라 함은 노드의 통신상대에 대한 정보나 노드의 상대적인 위치 및 거리에 대한 정보가 드러나지 않도록 하는 것을 의미하며 이는 곧 노드 상호간에 익명으로 라우팅 정보를 설정하고 통신하는 것을 말한다. 본 논문은 기존에 발표된 MANET 익명 라우팅 프로토콜을 분석하고 이를 기반으로 프라이버시를 보호하는 보다 효율적인 MANET 라우팅 프로토콜을 제안한다. 제안하는 기법은 기존의 방법보다 간단한 메시지 포맷과 적은 암호화 연산을 통해 더욱 효율적인 익명 경로 생성이 가능하다. 또한 효율성 분석에 있어서도 전송되는 메시지의 처리되는 특성을 분석하여 실질적으로 경로 설정 과정에 관계된 모든 노드들이 고려된 연산량 분석을 함으로써 보다 정확한 분석 결과를 제시한다.

본 논문은 서론을 포함하여 총 다섯 개의 장으로 이루어져 있다. 2장에서는 연구배경에 대해 설명하고 프라이버시 요구사항 및 관련 연구를 소개한다. 3장에서는 제안하는 프로토콜을 소개하며 4장에서 프로토콜의 익명성 및 효율성을 분석하고 5장에서 결론을 맺는다.

II. 연구배경

2.1 MANET의 프라이버시 요구사항 및 기 제안된 주요 기법

MANET에서는 소스노드와 목적노드가 상호간에 상대의 실제 ID를 알고 있다 그러므로 본 논문에서 말하는 프라이버시 보호는 소스노드와 목적노드간의 프라이버시 보호를 의미하는 것은 아니며 중간노드로부터 소스노드와 목적노드의 프라이버시를 보호하는 것과 라우팅 경로 밖의 노드로부터 경로상 노드의 프라이버시를 보호하는 것을 말한다.

2.1.1 MANET의 프라이버시 요구사항 및 기존 기법

프라이버시 보호 요구조건은 통신을 하는 과정에서 발생되는 ID정보, 위치정보, 경로정보의 노출을 방지하기 위한 요구사항과 하나의 메시지가 노출되었을 때 노출된 메시지와의 연관성으로 인해 다수 메시지의 ID, 위치, 경로 정보가 연쇄적으로 노출되는 것을 방지하기 위한 요구사항이 있다. MANET의 프라이버시 요구사항은 아래와 같이 나누어 볼 수 있다[10].

- ID 프라이버시: 통신하고자 하는 소스노드와 목적노드의 실제 ID를 라우팅 경로상의 중간노드들은 알 수 없고 그 반대도 마찬가지인 경우를 말하며 아래와 같이 세부적으로 나누어 볼 수 있다.
  - 소스노드 ID 프라이버시
  - 목적노드 ID 프라이버시
  - 중간노드 ID 프라이버시
- 위치 프라이버시: 소스노드와 목적노드의 상대적인 위치에 대한 정보의 보호를 의미하는 것으로서 약한 위치 프라이버시와 강한 위치 프라이버시로 나누어 볼 수 있다. 전자는 소스노드 및 목적노드까지 소요되는 거리(홉 수) 정보를 알 수 있으나 정확한 위치는 알 수 없는 경우를 말하며 후자는 소스노드와 목적노드까지의 거리에 관련된 어떤 정보도 알 수 없는 경우를 말한다. 본 논문에서는 GPS시스템이 적용되지 않는 ID기반 라우팅에 대해 이야기하고 있고 신호의 방향 등을 고려하지 않고 있음으로 지리적으로 정확한 위치의 개념이 논의될 수 없다. 그러므로 본 논문에서는 강한 위치 프라이버시만을 고려한다.
- 경로 프라이버시: 소스노드와 목적노드 사이에 설정된 경로에 대한 어떤 정보도 알 수 없는 경우를 말하며 아래와 같이 두 가지 세부항목으로 나누어진다.
  - 단일통신에 있어서의 홉 간 메시지의 비연결성: 경로상의 또는 경로 밖의 악의적 노드가 소스노드 또는 목적노드로 가는 패킷을 추적 할 수 없다.
  - 메시지 내의 경로정보 보호: 경로 밖의 악의적 노드는 전송되는 메시지를 통해 경로의 어떤 부분 정보도 알 수 없다.
- 세션간 통신에 있어서의 메시지 비연결성: 동일 노드에서 이전 세션과 현재 세션의 메시지는 상호간에 연결성이 없어야 함을 말하며 세부적으로 아래와 같이 나뉜다.
  - RREQ(Route Request)메시지와 RREP (Route Reply) 메시지의 비연결성

- 서로 다른 세션에서 동일노드가 전송한 메시지의 비연결성

2.1.2 MANET 프라이버시 보호를 위한 주요 기법

MANET에서는 프라이버시 보호를 위해 노드의 실제 ID가 노출되지 않아야 하며 이를 위해 소스노드와 목적노드의 IP 또는 MAC 주소를 사용하지 않고 메시지를 전송한다. 소스노드는 목적노드가 어느 위치에 있는지 알 수 없음으로 메시지를 플루딩하게 되고 이때 목적노드가 플루딩 되는 메시지를 확인 할 수 있는 방법이 필요하다. 트랩도어 기법은 전송 메시지에 목적노드의 ID대신 목적노드만이 식별할 수 있는 데이터(트랩도어)를 넣어 전송하는 방식으로 트랩도어 메시지를 받는 목적노드만이 자신에게 온 것임을 확인할 수 있고, 다른 노드들은 그에 대한 확인이 불가능하다. 일반적으로 목적노드와 사전 공유된 대칭키 또는 공개키를 이용하여 특정 메시지를 암호화함으로써 만들 수 있다. 그러나 이러한 방법은 목적노드가 아닌 노드들이라 할지라도 메시지를 받은 모든 노드가 그에 대한 확인을 위해 트랩도어에 대한 복호화를 시도해 보아야 하므로 오버헤드가 발생하며 특히 트랩도어가 공개키 방식으로 만들어져 있는 경우에는 오버헤드가 더욱 크다. Song 등이 제안한 AnonDSR[5]에서는 이러한 단점을 개선하여 익명ID 트랩도어를 이용한 2단계 기법을 제안한바 있다. 이 기법은 라우팅 경로 설정 단계 이전에 익명ID와 세션키를 확립하기 위한 단계를 수행함으로써 경로 설정 단계에서는 익명ID를 확인하는 간단한 과정만으로 메시지의 목적지를 확인할 수 있는 기법이다.

Goldschlag 등이 제안한 Onion[12] 기법은 중첩된 암호화를 통해 메시지를 전송한 노드가 출발노드인지 단순 중계노드인지 구분하기 어렵도록 하여 노드의 익명성을 제공하는 방법이다. 소스노드는 경로상 노드의 공개키를 이용해 데이터에 대한 중첩된 암호화를 수행함으로써 onion을 생성하고 onion을 수신한 노드들은 자신의 공개키로 한 겹씩 복호화하여 최종적으로 목적노드가 데이터를 획득하게 된다. 그러나 MANET 라우팅에서는 어떤 노드가 경로상의 노드인지 알 수 없고 공개키 암호화를 이용한 onion 생성으로 인한 오버헤드가 크기 때문에 onion 기법을 그대로 적용하기는 어렵다.

이와 관련하여 AnonDSR은 소스노드의 일회용 공개키를 이용하여 중간노드들이 onion 생성정보를 목적노드로 전달해 주는 방식으로 onion 기법을 MANET에 적용하고 있다. AnonDSR의 방법은 목적노드가 대칭키로 암호화된 onion을 생성하지만 플루딩으로 전달되는 RREQ(Route request)를 통해 각 홉에서 onion 생성을 위한 정보를 공개키를 이용하여 onion 형태로 전달함으로 오버헤드가 매우 크다. Kong과 Hong이 제안한 ANODR[4]에서는 부메랑 onion 기법을 이용하여 이와 같은 문제를 해결하고 있다. 부메랑 onion은 대칭키 방식으로 RREQ 과정에서 각 노드가 onion을 중첩하여 암호화함으로서 onion을 생성하고 RREP(Route reply)과정에서는 RREQ과정에서 만들어진 onion을 각 노드가 복호화 하여 검증하는 형태로서 경로상 노드의 키를 알아야 할 필요가 없으므로 효율적이다. 그러나 부메랑 onion 기법은 노드와 노드 사이의 RREQ메시지에 포함된 onion과 RREP메시지에 포함된 onion이 같으므로 RREQ와 RREP 메시지가 연결되는 문제점이 있다. ANODR에서는 각 노드의 일회용 공개키를 이용하여 홉 간 암호화를 함으로서 위의 문제를 해결하고 있다. 그러나 제안된 프로토콜을 분석해 보면 ANODR에서 onion은 메시지의 익명 식별을 위해 사용되고 있으나 일회용 공개키가 동일한 역할을 함으로 onion을 제거 하여도 익명성에는 아무런 문제가 없다.

2.2. 관련연구

MANET 라우팅에서의 프라이버시 요구 조건을 만족시키기 위한 여러 가지 제안들이 있었으며 본 장에서는 기 제안된 기법들에 대해 좀 더 자세히 살펴본다.

앞 절에서 언급된 ANODR에서는 트랩도어와 부메랑 onion을 사용하여 MANET에서의 프라이버시를 보호하는 기법을 제안하고 있다. 제안된 기법은 ID프라이버시와 경로프라이버시를 만족하고 세션간 메시지의 비연결성 또한 만족한다. 그러나 전송되는 onion 메시지의 규칙적인 증가로 인해 소스노드까지의 홉 거리가 노출될 수 있어 위치 프라이버시가 보호되지 못하는 문제가 있다. 그리고 대칭키를 이용한 트랩도어의 사용은 트랩도어 검증과정에서 다른 노드와 확립하고 있는 모든 키를 이용해 매 홉마다 반복적인 검증이 필요하므로 비효율적이며 앞 절에서 언급된 것과 같이 일회용 공개키 사용에 의해 onion의 사용이 불필요하다.

AnonDSR은 사전에 독립적인 익명ID 및 키 확립

과정을 둠으로서 실제 라우팅 경로 설정시의 연산량을 줄였다. 이 과정에서 소스노드와 목적노드는 사전에 공유된 공개키를 이용해 암호화된 통신을 하여 익명ID를 확립하고 확립된 익명 ID를 경로설정 단계에서의 트랩도어로 사용하여 트랩도어 확인을 위한 반복적인 암호화 연산을 제거하였다. 그러나 AnonDSR에서 사용하고 있는 onion 기법은 RREQ 전송시에 onion을 생성하기 위해 공개키 연산을 해야 하므로 익명ID에 의에 얻은 연산량의 이점을 다시 상쇄하고 있다. 그리고 AnonDSR의 onion 또한 규칙적으로 길이가 변화 하므로 소스노드의 유추가 가능하다. AnonDSR은 효율성 면에서는 좋지 않은 것이 사실이나 익명 ID의 사전 교환 방식은 실제 라우팅 경로 설정 단계에서의 연산량을 대폭 줄일 수 있는 방법이다

Seys와 Preneel 등의 ARM[6]은 소스노드와 목적노드간의 비밀키와 익명ID의 확립을 가정하고 있으며 실질적인 라우팅 경로 설정 과정은 AnonDSR과 유사하다고 볼 수 있다. ARM은 패딩 기법을 제안하고 있으며, 이것은 임의 길이의 메시지(패딩)를 onion 메시지에 넣어 메시지 길이의 규칙적 증가로 인한 소스노드의 위치 유추 문제를 해결하기 위한 기법이다. ARM에서는 RREQ 과정에서는 소스노드가 랜덤한 길이의 패딩을 onion에 추가하여 전송하고 RREP 과정에서는 패딩을 이용해 홉간 메시지의 길이를 동일하게 하는 방법을 제시하고 있는데, 분석 결과 RREP 과정에서 홉 간 메시지 길이를 동일하게 하지 않더라도 전송되는 onion에 랜덤 패딩을 삽입하는 것만으로 익명성은 보장된다. 패딩의 길이는 소스노드만 알고 있으므로 랜덤 패딩이 추가된 onion을 받은 노드는 길이를 이용해 onion을 전송한 노드가 소스노드인지 전달한 중간노드인지 알 수가 없으며, RREQ 과정에서 메시지 길이를 동일하게 하지 않더라도 패딩의 길이가 일정치 않음으로 onion을 받은 노드는 목적노드가 보낸 것인지 중계된 것인지 알 수 없고, 최소 길이의 onion도 결정지을 수 없으므로 소스노드 위치도 유추가 불가능하다. 이러한 패딩 기법은 ANODR이나 AnonDSR 기법에도 동일하게 적용될 수 있을 것으로 보인다. 그러나 ARM 또한 AnonDSR과 같이 RREQ 전송시에 onion 생성을 위해 공개키 암호화 연산을 하게 되므로 네트워크 전체적으로 상당히 큰 오버헤드가 발생한다.

Zhang 등의 MASK[7] 기법은 기존의 onion이나 트랩도어를 사용하지 않고 네트워크 참여시에 각 노드가 특정 크기의 검증가능한 일회용 익명ID(그룹키) 집합을 지급받는 형태로서 각각의 노드는 지급받은 그룹 키를 이용해 자신의 한 홉 거리에 있는 모든 노드와 익명 인증을 하여 각 노드에 대한 링크ID와 홉 간 세션키를 확립한다. 그러나 한 홉 거리 노드와의 익명 인증은 그 자체로 상당히 비효율적이며, 네트워크 참여시에 모든 노드가 다수의 그룹키를 지급받는다는 가정 또한 현실적이지 못하다. 뿐만 아니라 MASK는 RREQ메시지에서 목적노드의 ID를 숨기지 않으므로 목적노드가 그대로 노출되는 문제가 있다. MASK에서는 RREQ메시지를 네트워크 전체 플루딩하여 ID와 목적노드를 연결 지을 수 없으므로 문제되지 않는다고 주장하고 있으나 앞장에서 제시된 익명성 기준에 의하면 목적노드 ID의 노출 자체로 이미 ID 프라이버시를 만족하지 못하고 있는 것으로 볼 수 있다.

Lu 등이 제안한 ASRPAKE[8]와 Shokri 등이 제안한 PuesdoAODV[9], Shao와 Huang이 제안한 TEAR[11] 또한 익명 라우팅 경로 설정 방법을 제안하고 있다. 그러나 제안된 기법들은 모두 메시지 비연결성을 만족하지 못하며, ASRPAKE와 TEAR은 중간노드의 ID도 노출된다. 효율성 측면에 있어서도 트랩도어 확인 또는 onion의 생성을 위한 반복적인 암호화 연산이 요구되므로 비효율적이다. TEAR은 AnonDSR과 유사한 형태의 라우팅 프로토콜을 가지며 랜덤 패딩 등을 고려하지 않은 onion의 사용으로 인해 이전에 제시된 논문들과 마찬가지로 소스노드의 홉 거리 유추가 가능하다.

III. 제안하는 프로토콜

본 장에서는 onion 기법을 사용하지 않고 익명성을 보장하는 효율적인 MANET 익명 라우팅 프로토콜을 제안한다. 제안하는 프로토콜은 onion 생성이나 인증 과정에서 추가로 요구되는 암호화 연산을 제거하여 보다 효율적으로 라우팅경로의 설정이 가능하다. 그리고 onion을 사용하지 않으므로 소스노드의 유추 문제와 이를 막기 위한 기법에 대한 고려를 근본적으로 해결한다. 또한 AnonDSR에서 제안되었던 익명 ID 사전 확립 기법을 도입하고 AnonDSR과는 달리 익명 ID의 자체갱신 기법을 제안함으로써 효율성을 높였다. 결과 분석 과정에서도 기존의 논문들은[5,6] 경로상의 노드에 국한된 연산량 분석을 수행 하였으나 제안하는 논문은 경로 확립에 참여하는 모든 노드의

[표 1] 표기법

표기법	내 용
N_i	노드 i (i=A or B or C) NS: 소스노드, ND: 목적노드
ID_i	노드 i의 실제 ID
PID_i	노드 i의 익명 ID
LID_{ij}	노드 i와 j사이의 링크ID
SK_{ij}	소스노드 i와 목적노드 j의 세션키
K_{ij}	노드 i와 j의 링크간 세션키
K_i	노드 i가 생성한 임의의 키
$\pm TPK_i$	노드 i가 생성한 일회용 공개키 쌍 (+TPK: 공개키, -TPK: 개인키)
$\pm K_i$	노드 i의 공개키 쌍 (+K: 공개키, -K: 개인키)
$\{\ \}_K$	키 K를 이용한 암호화 연산
$MAC_K(\ )$	키 K를 이용한 MAC함수
$H(\ )$	해시함수
R_i	노드 i가 생성한 랜덤 값
C_Q	RREQ 플루딩 처리시의 연산량
C_P	RREP 메시지 처리시의 연산량
C_U	경로상 노드의 메시지 처리시 연산량

연산량을 고려하여 분석함으로써 더욱 정확한 효율성 분석을 수행 하였다.

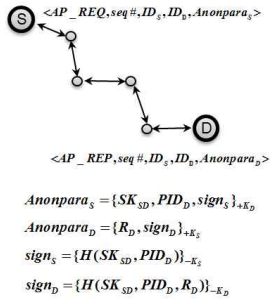
제안하는 프로토콜은 아래와 같은 가정을 사용한다.

- 소스노드가 자신이 통신하고자 하는 목적노드의 인증된 공개키를 보유하고 있다.
- 네트워크는 대칭형으로 인접한 이웃노드는 상호간에 통신이 가능하다.

프로토콜은 익명 파라미터 확립 단계, 익명 라우팅 경로 설정 단계, 익명 데이터 통신단계의 세 부분으로 나누어지며 사용되는 표기법은 [표 1]과 같다.

3.1 익명 파라미터 확립 단계

익명 파라미터 확립 과정은 소스노드와 목적노드의 실제 ID가 공개된 상태에서 이루어지며 이 과정을 통해 소스노드와 목적노드는 상호간에 사용할 익명 ID와 세션키를 확립하게 된다. 이 단계는 AnonDSR 에서 제안한 것과 유사한 방법이다. 또한 이 단계는 같은 소스노드와 목적노드간에는 오직 한번만 수행된다. 소스노드는 익명 파라미터의 확립을 위한 메시지를 [그림 1]



[그림 1] 사전 익명 파라미터 확립단계

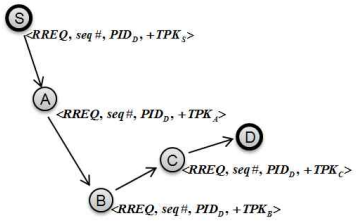
과 같이 생성하여 전송한다. 제시된 [그림 1]은 익명 파라미터 확립을 위한 메시지 전송 단계를 자세히 나타내고 있다.

소스노드가 전송하는 메시지 내의 AP_REQ는 익명 파라피터 요청 메시지임을 나타내는 값이고 seq#는 플루딩되는 메시지의 중복을 제어하기위한 파라미터이다. 소스노드와 목적노드의 ID가 메시지에 공개되어 있음으로 중간노드는 ID를 확인하여 신속하게 메시지를 전송하며 메시지를 받은 목적노드는 $Anonpara_S$를 복호화하여 소스노드와 사용할 익명 ID와 세션키를 획득한다. 메시지 상의 sign값은 인증을 위한 서명값으로 소스 및 목적노드는 이 값을 검증하여 상호간에 인증한다. AP_REQ 메시지를 받은 목적노드는 그에 대한 응답으로 AP_REP 메시지를 플루딩하고 소스노드는 $Anonpara_D$를 복호화하여 서명을 검증함으로써 익명ID의 확립이 정상적으로 이루어졌음을 확인한다.

3.2 익명 라우팅 경로 설정 단계

익명 라우팅 경로 설정 과정은 이전 단계에서 확립된 익명 ID를 이용하여 진행된다. 소스노드는 [그림 2]에서와 같이 RREQ 메시지를 생성하여 네트워크에 플루딩 한다.

RREQ 메시지를 받은 중간노드는 먼저 seq#를 확인하여 중복된 메시지인지 여부를 판단하여 중복될 시에는 메시지를 무시한다. 메시지가 중복되지 않는 경우 중간노드는 PID를 통해 자신이 목적노드인지 아닌지를 확인하여 자신이 목적노드가 아닌 경우에는 라우팅 테이블에 <PIDD, +TPKi-1, null, ±TPKi, null>을 저장하고 RREQ 메시지 내의 일회용 공개키를 자신이 생성한 일회용 공개키로 변경하여 다시 플



[그림 2] 라우팅 경로 설정 단계(RREQ전송)

루딩 한다.

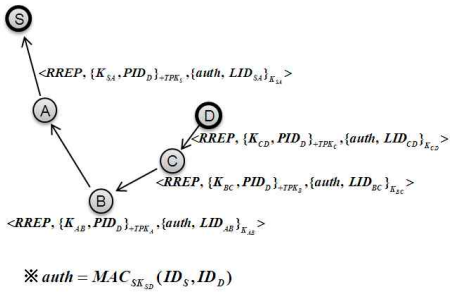
라우팅 경로가 [그림 2]와 같이 설정된다고 가정할 때 RREQ 과정에서 노드 B는 [표 2]와 같은 라우팅 테이블을 유지하게 된다.

[표 2] RREQ 과정에서 노드 B가 유지하는 라우팅 테이블

목적 노드의 익명 ID	일련번호	이전 노드의 공개키	이전 노드와의 링크정보	자신이 생성한 공개키쌍	다음 노드와의 링크정보
...	...	...	...	...	...
PIDD	seq#	+TPKA	null	±TPKB	null
...	...	...	...	...	...

라우팅 테이블에서 이전 노드와의 링크 정보와 다음 노드와의 링크정보는 RREQ 과정에서 알 수 없고 RREP 과정에서 채워지게 된다. RREQ 메시지를 받은 목적노드는 [그림 3]과 같이 RREP 메시지를 생성하여 전송한다.

RREP 메시지 내의 auth는 인증값으로 소스노드는 최종적으로 이 값을 확인하여 실제 목적노드가 RREP 메시지를 전송하였음을 알 수 있다. 메시지를 받은 중간노드는 메시지의 두 번째 요소를 복호화하여 PID가 자신의 라우팅 테이블에 있는지 확인함으로써 자신이 경로상의 노드인지 아닌지를 판단하게 된다.



[그림 3] 라우팅 경로 설정 단계(RREP전송)

경로상의 노드가 아닌 경우에는 RREP의 포워딩을 중지하며 경로상의 노드임이 판명되면 RREP 메시지를 참고하여 라우팅 테이블을 완성하게 된다. RREP 과정에서 노드 B가 유지하는 라우팅 테이블을 표 3.에 나타내었다.

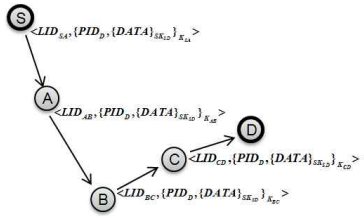
[표 3] RREP 과정에서 노드 B가 유지하는 라우팅 테이블

목적 노드의 익명 ID	일련 번호	이전 노드의 공개키	이전 노드와의 링크정보	자신이 생성한 공개키쌍	다음 노드와의 링크정보
...	...	...	...	...	...
PIDD	seq#	+TPKA	<LIDAB, KAB>	±TPKB	<LIDBC, KBC>
...	...	...	...	...	...

경로상의 노드는 다음 홉의 노드와 공유할 링크 아이디 및 링크 세션키를 라우팅 테이블 내 이전 홉 노드의 일회용 공개키를 이용하여 암호화하고 이를 이용하여 RREP 메시지를 갱신한 후 다시 RREP 메시지를 전송한다. RREP 메시지를 받은 소스노드는 auth를 검증하여 RREP 메시지가 목적노드에게서 온 것인지 확인하고 라우팅 테이블을 완성한다. RREP 메시지는 각 노드가 생성한 홉 간 세션키에 의해 암호화되고 해당 키는 이전 홉 노드의 일회용 공개키로 암호화되므로 일회용 공개키를 발급한 노드만이 RREP 메시지를 복호화 하여 올바른 메시지를 획득할 수 있다. [그림 3]에서 노드 B는 노드 C로부터 RREP를 수신하여 자신이 경로상의 노드임을 확인하고 RREP를 다시 중계하고 노드 A가 RREP를 수신하여 자신이 경로상의 노드임을 확인하고 다시 중계한다. 이 과정에서 노드 B는 노드 C와 A의 전송영역 내에 있고 RREP의 메시지 비연결성으로 인해 중복확인이 불가능하므로 두 노드가 보낸 RREP를 모두 처리해야 한다.

3.3 익명 데이터 통신

익명 경로 설정이 끝나면 소스노드는 확립된 한 홉 거리 노드와의 링크ID와 목적노드의 익명ID를 이용하여 익명 데이터전송이 가능하다. 소스노드는 [그림 4]와 같이 메시지를 생성하여 전송한다. 메시지를 받은 중간노드는 라우팅 테이블 내의 링크ID를 확인하여 자신이 처리해야할 메시지인지 여부를 판단한다. 경로상의 중간노드는 메시지를 복호화 한 후 익명ID를 확인하여 자신이 목적노드인지 알 수 있으며 목적노드가



[그림 4] 익명 데이터 전송단계

아닐 경우 라우팅 테이블을 참조하여 자신의 다음 홉 노드와 확립된 링크 세션키와 링크ID를 이용하여 메시지를 갱신한 후 변경하여 전송한다.

3.4 익명 ID의 갱신

라우팅경로 설정 단계에서 사용되는 익명ID는 각 세션마다 새로운 값으로 갱신되어야 할 필요가 있다. 익명ID가 갱신되지 않으면 서로 다른 세션에서 전송되는 메시지 내에 동일한 익명ID값이 포함되므로 세션 간 메시지가 연결되고 메시지 비연결성 조건을 만족하지 못한다. AnonDSR과 같은 경우 경로설정 과정에서 다음 세션에 사용할 익명 ID를 메시지에 포함하여 제공 하는 방식으로 갱신을 제안한 바 있으나 본 논문에서는 노드 상호간 공유된 키 값을 이용한 자체 갱신 기법을 제안한다. 자체갱신 기법과 전송을 통한 갱신 기법에 대한 비교 결과는 [표 4]와 같다.

비교된 두 기법 모두 익명ID의 동기화를 위해 현재의 익명 ID이외에 이전 세션의 익명ID를 추가적으로 유지해야 할 필요가 있다. 이는 [그림 5] 및 [그림 6]의 알고리즘에서 설명하고 있는 것과 같이 ID갱신을 위한 과정에서의 메시지가 차단되거나, 손실 되는 경우 익명 ID의 동기화를 위해 필요하다. 익명ID가 동기화 되지 않으면 매번 통신시마다 3.1절의 과정을 거쳐 익명 ID를 확립해야 하므로 3.1절을 수행하는 비용이 추가된다.

비교 결과에서 알 수 있듯이 두 기법은 익명ID의 동기화를 위해 요구되는 정보 및 키 노출에 따른 문제점 측면에서는 동일한 반면 자체갱신 기법이 상대적으로 짧은 메시지의 길이를 가지게 됨으로 무선 네트워크에 의한 데이터 전송에 있어서 보다 효율적이라 할 수 있다. 익명ID를 갱신하는 과정에서 노드 상호간에 갱신되는 익명ID는 동기화 할 필요가 있으며 제안된 기법은 [그림 5] 및 [그림 6]과 같은 알고리즘을 이용하여 익명ID를 동기화 한다.

알고리즘상의 C_PID와 C_KEY는 현재의 익명ID와 세션키를 의미하고 P_PID와 P_KEY는 이전 세션의 익명ID와 세션키를 의미한다. 소스노드는 정상적인 RREP메시지를 받았을 경우에만 익명ID의 갱신을 수행하며 목적노드는 RREQ메시지에서 사용된 익명ID를 확인하여 정상적으로 현재 세션의 익명ID를 사용하고 있으면 갱신을 수행하고, 이전 세션의 익명ID를 사용하고 있으면 현재의 익명ID를 이용하여 응답한다. 세션키의 자체 갱신은 현재의 세션키에 대한 해시 연산을 통해 이루어지고, 익명ID의 자체갱신은 갱신된 세션키를 이용한 MAC연산을 통해 이루어진다. 이와 같이 자체 갱신을 하게 되면 키를 알지 못하는 악의적인 공격자는 현재 익명ID를 이용하여 다음 익명ID나 이전의 익명ID를 계산 할 수 없다. 노드의 현재 상태가 노출된 경우에는 제시된 두 기법 모두 동일하게 노드가 저장하고 있는 한 세션 이전의 익명ID만이 노출되며 그 이전의 익명 ID나 세션키는 알 수 없다. 따라서 자체갱신 기법은 익명ID에 의한 세션간 메시지의 연결 불가능성을 만족하고, 노드의 상태 노출의 경우에도 익명ID 전송의 방법과 동일한 정도의 세션간 데이터 비연결성을 제공한다. 노드가 노출된 경우그 이후의 익명 ID와 세션키는 계산가능하나 이는 노드가 이미 노출된 것이므로 본 논문에서는 고려하지 않는다.

[표 4] 익명 ID갱신 기법의 비교

구분	다음세션 익명ID를 암호화하여 전송	자체갱신
익명ID의 동기화를 위해 요구되는 정보	두 기법 모두 현재의 익명ID와 이전 세션의 익명ID를 유지해야 함	
키 노출에 따른 문제점	익명ID를 암호화한 세션키가 노출 되어도 이전 세션의 익명 아이디는 알 수 없다	자체 갱신을 위한키가 노출되어도 일방향 함수(MAC)에 의해 키가 갱신되므로 이전 세션의 익명ID를 알 수 없다
메시지 길이	상대적으로 메시지 길이가 길다	상대적으로 메시지 길이가 짧다

```
Send RREQ(C_PID)

receiveRREP( ) {
    If(collect RREP) {
        P_KEY=C_KEY
        C_KEY=H(C_KEY)
        P_PID=C_PID
        C_PID=MAC_C_KEY(C_PID)
    }
}
```

[그림 5] 소스노드의 익명 ID 갱신 알고리즘

```
receive collect RREQ( ) {
    if(RREQ.PID==C_PID) {
        P_KEY=C_KEY
        C_KEY=H(C_KEY)
        P_PID=C_PID
        C_PID=MAC_C_KEY(C_PID)
        sendRREP(C_PID)
    }
    else if(RREQ.PID==P_PID) {
        sendRREP(C_PID)
    }
    else {
        forwardRREQ( )
    }
}
```

[그림 6] 목적노드의 익명 ID 갱신 알고리즘

IV. 익명성 및 효율성 분석

본 장에서는 제안된 기법과 기 제안된 기법을 비교 분석한다. 앞 장의 관련연구를 통해 여러 가지 기 제안된 기법을 살펴보았으나 ARM과 같은 경우에는 AnonDSR과 유사한 면이 많고, ASRPAKE와 PuesdoAODV, TEAR등은 최근에 제안되었지만 많은 부분에서 익명성 요구조건을 만족하지 못하고 비효율적 이므로 비교대상에서 제외하였다.

4.1 익명성 분석

제안하는 프로토콜은 사전 익명ID 확립 단계(1단계)에서 노드의 실제 ID를 사용하지만 다음 단계인 익명 라우팅 경로 설정 단계(2단계)와 연결 되지 않으므로 실제로는 소스노드와 목적노드가 언젠가 통신할 가능성이 있다는 것은 알 수 있지만 2단계 과정에서 실제로 어떤 노드가 통신을 하고자 하는지는 알 수 없다. 그러므로 1단계의 실제 노드ID 공개는 프라이버시 노출에 아무런 영향을 미치지 못한다.

제안된 프로토콜은 소스노드의 ID를 사용하지 않고 익명ID를 트랩도어로 사용하여 익명 통신을 하고자 하는 노드의 실제 아이디가 드러내지 않는다. 경로상 중간노드의 ID 또한 사용되지 않고 발급된 일회용 공개키로 메시지를 인증하기 때문에 중간노드의 ID 익명성도 만족한다. 위치 프라이버시 보호 측면에서는 onion 등과 같이 홉 정보를 유추 할 수 있는 어떤 정보도 RREQ 또는 RREP 메시지에서 사용하고 있지 않으며 항상 동일한 크기의 메시지를 사용하므로 소스노드와 목적노드의 홉 정보는 드러나지 않는다. 또한 전송되는 RREP 메시지와 데이터는 홉 간 세션키에 의해 암호화 되므로 연결되지 않으며 각각의 노드가 경로상의 이전노드 및 다음노드와의 링크 ID만 유지할 뿐 경로정보는 메시지에 포함되지 않음으로 경로 프라이버시 또한 보호된다. 제안된 프로토콜의 라우팅 경로 설정 단계의 메시지는 여러 세션 간에도 비연결성을 만족함으로 세션간 통신에 있어서의 메시지 비연결성도 만족한다. 프라이버시 보호 요구조건에 대한 만족도 비교 결과는 [표 5]와 같다. ANODR과 AnonDSR은 onion의 규칙적인 길이 변화로 인해 소스노드의 홉 거리가 유출 될 수 있는 문제가 있으나 2.2절의 관련연구 부분에서 분석한 것과 같이 단순패딩 추가를 통해 해결될 수 있으므로 △로 표시하였다.

4.2 효율성 분석

프로토콜의 효율성은 라우팅 경로 설정시에 소요되는 연산량을 비교하여 분석할 수 있다. 그러나 본 논문은 라우팅 경로설정 단계 이전에 익명ID 확립을 위한 사전 단계를 수행하므로 이에 대한 고려가 필요하다. 익명 파라미터 확립 과정의 비용 또한 효율성 분석에 포함될 수 있다면 좀 더 정확한 분석이 될 것이나 아래와 같은 분석 결과에 따라 이 비용은 무시한다.

- 기 제안된 논문 중에는 사전 익명 파라미터 확립과정을 가정하고 진행하는 경우가 있다. 실제로 익명 파라미터의 확립 과정에서 어떤 방식을 사용하는지에 대한 언급이 없음으로 비교가 불가능 함.
- 연산량 측면에서 보면 현재 제안된 논문에서의

[표 5] 프라이버시 보호 요구조건 만족도 비교

요구조건			ANODR	AnonDSR	제안하는 프로토콜
ID 프라이버시 보호	소스노드ID		O	O	O
	목적노드ID		O	O	O
	중간노드ID		O	O	O
Location 프라이버시 보호			△ (페딩 기법 이용가능)	△ (페딩 기법 이용가능)	O
Route 프라이버시 보호	단일통신에서의 홉간 메시지 비연결성	RREP 메시지	O	O	O
		전송 데이터	O	O	O
	경로정보의 보호		O	O	O
세션간 통신에 있어서의 메시지 비연결성	RREQ-RREP		O	O	O
	서로 다른 세션에서 동일노드가 전송한 메시지의 비연결성		O	O	O

사전 익명 파라미터 확립 과정의 연산량은 최종 분석된 연산량의 복잡도에 영향을 미칠 정도는 아님.

• 전송비용 측면에서 2단계 통신을 함으로써 발생되는 추가적인 통신비용과 2단계 통신을 하지 않고 단일 단계 라우팅 프로토콜을 진행함 따라 증가되는 연산 비용 사이의 비교가 필요함. 익명 파라미터 확립을 위한 추가적인 통신은 네트워크 초기에 1회만 일어나는 반면 단일 단계 라우팅 프로토콜을 진행함에 따른 증가된 연산비용은 라우팅 경로설정 단계가 일어 날 때마다 발생하게 됨. 이동성을 특징으로 가지는 MANET 환경에서는 라우팅 경로 설정 단계가 빈번히 일어나므로 2단계 통신을 이용한 기법이 보다 효율적일 것으로 생각됨.

또한 제안하는 기법과 ANODR 기법은 일회용 공개키의 생성비용이 요구되나 이는 사전에 생성할 수 있고 프로토콜 진행시에는 영향을 미치지 않음으로 고려하지 않는다. MANET 익명 라우팅의 경로설정과정에서 일어날 수 있는 메시지 처리 형태에 따라 연산량을 구분하면 아래와 같이 나누어 볼 수 있다.

• RREQ 플루딩 처리시 전체 네트워크에서 소요되는 연산량(C_Q): RREQ 메시지는 플루딩 형태로 전송되므로 메시지를 수신한 노드는 최소 한번 이를 처리해야 하며 메시지 내의 일련번호에 의해 중복된 메시지의 처리는 막을 수 있다.

• RREP 메시지 처리시 전체 네트워크에서 소요되는 연산량(C_P): RREP는 플루딩 형태로 전송되지는 않지만 이 메시지를 수신한 모든 노드는 이를 처리해야 하며 이중 경로상의 노드만 RREP를 중계한다. 3장의 RREP 전송 부분에서 언급된 것과 같이 RREP 메시지는 중복으로 처리될 수 있다.

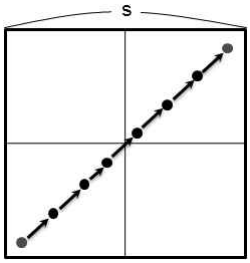
• 경로상 노드의 메시지 처리 연산량(C_U): 경로상의 노드들만 메시지를 처리하기 위한 연산이 필요한 경우

연산량의 분석은 C_Q, C_P, C_U의 연산량 차이를 비교한 후 각 프로토콜에서 소요되는 연산량을 C_Q, C_P, C_U로 표현하여 프로토콜 간 연산량의 차이를 비교한다. 분석을 위해 노드는 특정 범위($s \times s$)의 공간에 균일하게 분포되어 있다고 가정하며 아래와 같이 관련 요소를 정의 한다.

• s: 노드 분포 영역의 가로, 세로 길이
• t: 노드의 최대 전송 거리
• d: 노드 밀도
• h: 메시지 전달 홉 수

보통 RREQ를 보낼 때 TTL(Time-To-Live)를 사용하여 플루딩 되는 영역을 제한할 수 있지만 익명 라우팅의 경우 TTL 정보가 위치 프라이비시에 나쁜 영향을 주므로 이 논문에서는 사용하지 않는다고 가정한다. 따라서 네트워크에 참여하는 모든 노드는 RREQ 메시지를 받게 된다. 반면에 RREP는 소스노드와 목적노드 경로 상에 있는 노드들과 그 주변 노드들만 수신하게 된다. 따라서 RREP는 RREQ에 비해 상대적으로 적은 노드가 참여하게 된다. 직관적으로 고려하여도 소스노드와 목적노드 간에 홉 수가 적으면 적을수록 RREP에 참여하는 노드는 적어진다.

RREQ는 소스노드와 목적노드의 위치와 관계없이 항상 $C_Q = ds^2$ 노드가 메시지를 수신하여 처리하게 된다. RREP의 경우에는 [그림 7]에 제시된 경우가 소스노드와 목적노드 간에 거리가 가장 긴 경우이며 소스



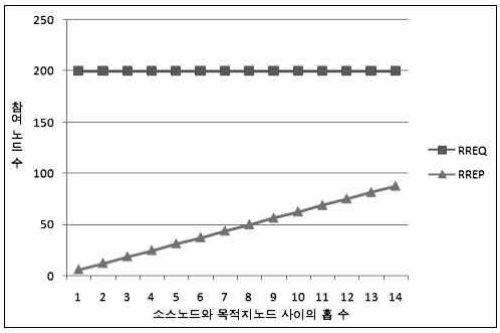
[그림 7] 분석 환경

노드와 목적노드 사이의 홉 수가 h일 때 $C_P = d\pi t^2 h$로 나타낼 수 있다. 경로상 노드의 메시지 처리시 연산량은 경로상의 각 노드가 한 번씩 연산하는 경우 이므로 $C_U = h$로 표현 할 수 있다. 예를 들어 $s = 1000$, $t = 100$으로 가정하고 200개의 노드가 균일하게 분포되어 있다고 하면 밀도 $d = 1/5000$로 계산된다. 가정에 의해 가로, 세로의 홉 수는 10홉이 되고 그림 7. 과 같은 최악의 경우 홉 수 $h = 10\sqrt{2} \simeq 14$가 된다. RREP의 홉 수 변화에 따라 RREQ와 RREP간의 연산량을 비교해 보면 [그림 8]과 같다.

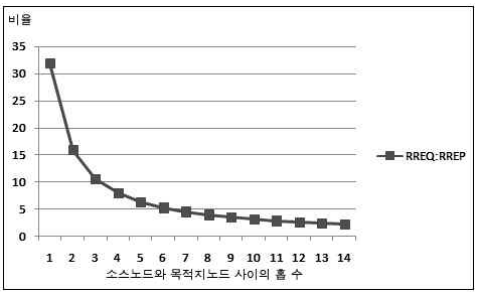
RREQ처리시의 연산량(C_Q)의 경우 전체 네트워크 플루딩 되어야 하므로 홉 수 와 관계없이 항상 200개의 노드가 RREQ메시지를 처리하게 된다. 그러나 RREP처리시의 연산량(C_P)과 같은 경우는 제시된 식에 따라 계산하면 홉 수 h에 비례하여 연산에 참여하는 노드수가 증가한다. [그림 8]의 그래프를 보면 RREP메시지의 처리시에는 평균적으로 50여개의 노드가 연산을 수행하며, 최대 RREP홉 수인 경우에도 약 100여개의 노드만이 연산에 참여하여 네트워크 전체적으로 볼 때 RREQ메시지 처리시의 연산량(C_Q)보다 월등히 적음을 알 수 있다.

소스노드와 목적노드 사이의 홉 수에 따른 연산량의 차이를 비율로 나타내면 [그림 9]와 같다. RREP의 홉 수가 적을 때는 연산량의 차이가 크며 최대의 홉 수일 경우에도 2배 이상의 차이를 보인다.

프로토콜의 연산량을 비교해 보면 ANODR 기법은 RREQ 전송시에 공개키 연산이 사용되지 않으나 대칭키 방식의 트랩도어와 onion의 사용으로 인해 많은 대칭키 연산이 필요하다. RREP 전송시에는 발급된 일회용 공개키에 의해 홉간 암호화를 제공하므로 공개키 연산이 요구된다. 그리고 RREP메시지의 확인을 위해서 모든 노드는 onion 검증을 해야 하므로 대칭키 연산에 있어서도 C_P의 연산량이 요구된다. AnonDSR은 RREQ 전송과정에서 트랩도어 확인을 위



[그림 8] 소스노드와 목적노드 사이의 홉 수에 따른 참여노드수(연산량)의 비교



[그림 9] 소스노드와 목적노드 사이의 홉 수에 따른 참여노드수(연산량)의 비율

한 비용은 소요되지 않으나 onion 생성을 위한 정보가 공개키를 이용한 방식으로 암호화 되므로 공개키 연산에 있어서 C_Q의 연산량이 요구된다. RREP과정에서는 공개키 연산이 사용되지 않으며 onion의 확인을 위해 대칭키 연산에 있어서만 C_P의 연산량이 요구된다. 제안하는 프로토콜은 RREQ과정에서 공개키를 사용하지 않고 onion을 사용하지 않음으로 암호화 연산이 요구되지 않는다. RREP과정에서는 일회용 공개키를 사용하므로 노드들이 RREP를 확인하기 위해 C_P의 공개키 연산이 요구되며 대칭키 연산에 있어서는 경로상 노드의 홉 간 암호화를 위한 비용 밖에 들지 않는다. 기 제안된 프로토콜과 제안하는 기법의 연산량을 비교하면 [표 6]과 같다.

비교 결과 공개키 연산량 측면에서는 ANODR과 제안하는 프로토콜이 C_P의 연산량을 가지므로 비슷하나 AnonDSR은 C_Q의 연산량을 가지므로 효율성이 떨어진다. 대칭키 연산 측면에서는 제안하는 프로토콜이 C_P의 연산량을 가지므로 C_Q의 연산량을 가지는 ANODR, AnonDSR보다 현저하게 효율적인 것을 알 수 있다.

[표 6] 효율성 비교

구분		공개키연산		대칭키연산	
		항목별	합계	항목별	합계
ANODR	RREQ	0	CP+CU	m*CQ+CQ	(m+1)CQ+2CP+CU
	RREP	CP+CU		2CP+CU	
	목적노드	0		0	
Anon DSR	RREQ	CQ	CQ+CU	CQ	CQ+CP+2CU
	RREP	0		CP	
	목적노드	CU		2CU	
제안하는 프로토콜	RREQ	0	CP+CU	0	2CU
	RREP	CP+CU		2CU	
	목적노드	0		0	

※ m: 소스노드가 네트워크상의 다른 노드와 확립하고 있는 키 개수

V. 결 론

제안하는 기법은 익명 ID를 이용하여 MANET에서 보다 효율적인 방법으로 익명라우팅 경로를 설정하고 익명으로 데이터 통신을 수행하였다. 기존의 제안된 논문과 비교해 볼 때 제안된 기법은 onion을 사용하지 않음으로 추가적인 암호화 연산과 onion으로 인해 발생할 수 있는 익명성의 침해 요인을 원천적으로 제거하였다. 또한 기 제안된 논문의 익명ID 트랩도어 기법을 도입하고 익명ID자체 갱신 기법을 제안함으로써 더욱 효율적인 방법으로 개선하였다. 분석에 있어서도 기존의 연산량 분석은 경로상의 각 노드에 국한된 분석을 수행하였으나 본 논문은 라우팅 경로 확립에 참여하는 모든 노드의 연산량을 고려하여 분석함으로써 더욱 정확한 효율성 분석이 가능하였다. 이 분석을 통해 공개키 연산 측면에서는 기존 기법중 가장 우수한 ANODR과 유사하나 ANODR은 onion 사용에 따른 추가 비용이 소요되므로 전체 네트워크 소요 비용 측면에서 본 기법이 가장 우수하다.

참 고 문 헌

[1] C.E. Perkins and P. Bhagwat, “Highly dynamic Destination Sequenced Distance-Vector routing (DSDV) for mobile computers,” ACM SIGCOMM Computer Communication Review, vol. 24, no. 4, pp. 234-244, Oct. 1994.

[2] D. Johnson, Y. Hu, and D. Maltz, “The Dynamic Source Routing Protocol(DSR) for Mobile Ad Hoc Networks for IPv4,” IETF RFC 4728, Feb. 2007.

[3] C.E. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector(AODV) Routing,” IETF RFC 3561, July 2003.

[4] J. Kong and X. Hong, “ANODR: Anonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks,” 4th ACM international symposium on Mobile ad hoc networking & computing, pp. 291-302, June 2003.

[5] R. Song, L. Korba, and G. Yee, “AnonDSR: Efficient Anonymous Dynamic Source Routing for Mobile Ad-Hoc Networks,” 3rd ACM workshop on Security of ad hoc and sensor networks, pp. 33-42, Nov. 2005.

[6] S. Seys and B. Preneel, “ARM: Anonymous Routing Protocol for Mobile Ad Hoc Networks,” 20th International Conference on Advanced Information Networking and Applications, pp. 133-137, Apr. 2006.

[7] Y. Zhang, W. Liu, W. Lou, and Y. Fang, “MASK: Anonymous On-Demand Routing in Mobile Ad Hoc Networks,” IEEE Transactions on wireless communications, vol. 5, no. 9, pp. 2376-2385, Sep. 2006.

[8] R. Lu, Z. Cao, L. Wang, and C. Sun, “A secure anonymous routing protocol with authenticated key exchange for ad hoc networks,” Computer Standards & Interfaces, vol. 29, no. 5, pp. 521-527, July 2007.

[9] R. Shokri, M. Yabandeh, and N. Yazdani, “Anonymous Routing in MANET using Random Identifiers,” Sixth International Conference on Networking, p. 2, Apr. 2007.

[10] B. Zhu, Z. Wan, M.S. Kankanhalli, F. Bao, and R.H. Deng, “Anonymous Secure Routing in Mobile Ad-Hoc Networks,” 29th Annual IEEE International Conference on Local Computer Networks, pp. 102-108, Nov. 2004.

[11] M. Shao and S. Huang, “Trust Enhanced Anonymous Routing in Mobile Ad-Hoc Networks,” Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 335-341, Dec. 2008.

[12] D.M. Goldschlag, M.G. Reed, and P.F. Syverson, “Hiding Routing Information,” First International Workshop on Information Hiding, Lecture Notes in Computer Science, 1174, pp. 137-150, May 1996.

<著者紹介>

이 승 윤 (Sung-yun Lee) 학생회원
2007년 2월: 한양대학교 전자컴퓨터공학부(학사)
2009년 8월: 한양대학교 컴퓨터공학과(석사)
2009년 7월 ~ 현재: 이글루시큐리티 인터넷보안연구소 연구원
<관심분야> 네트워크 보안, 프라이버시 보호

오 희 국 (Heekuck Oh) 종신회원
1983년: 한양대학교 전자공학과(학사)
1989년: 아이오와주립대학 전자계산학과(석사)
1992년: 아이오와주립대학 전자계산학과(박사)
1993년 ~ 1994년: 한국전자통신연구원 선임연구원
1995년 3월 ~ 현재: 한양대학교 컴퓨터공학과 교수
<관심분야> 암호프로토콜, 네트워크 보안
URL:http://infosec.hanyang.ac.kr/~hkoh/

김 상 진 (Sangjin Kim) 종신회원
1995년 2월: 한양대학교 전자계산학과(학사)
1997년 2월: 한양대학교 전자계산학과(석사)
2002년 8월: 한양대학교 전자계산학과(박사)
2003년 3월 ~ 현재: 한국기술교육대학교 인터넷미디어공학부 부교수
<관심분야> 암호기술 응용
URL:http://infosec.kut.ac.kr/sangjin/