

Mobile IPv4/Mobile IPv6 통신을 위한 모바일 헤더 변환 메커니즘

준회원 김대선*, 정회원 홍충선**

A Mobility Header Conversion Mechanism for Mobile IPv4 and Mobile IPv6 Communications

Dae-Sun Kim* *Associate Member*, Choong-Seon Hong** *Regular Member*

요 약

IPv4/IPv6 통신 서비스를 위한 변환기는 호스트의 변경 없이 IPv6 호스트와 IPv4 호스트간의 통신을 지원해 준다. 그러나 호스트가 모바일 노드일 경우 삼각형 라우팅 문제가 발생한다. Mobile IPv4 모바일 노드가 이동할 경우 홈 에이전트와 IPv6의 CN(Correspondent Node)에게 반드시 이동 정보를 보내 주어야 하지만 현재 NAT-PT는 IPv4와 IPv6의 이동성 헤더인 모바일 IPv4/Mobile IPv6의 헤더 변환은 지원하지 않기 때문에 삼각형 라우팅 문제가 발생하여 Mobile IPv4와 Mobile IPv6간에 비효율적인 통신을 일으킨다. 본 논문에서는 Mobile IPv4와 Mobile IPv6 통신에서 발생하는 삼각형 라우팅 문제를 해결하기 위한 모바일 헤더 변환 메커니즘을 제안하였다.

Key Words : NAT-PT, IPv6, IPv4, Mobile IPv6, Mobile IPv4

ABSTRACT

An interworking translator for IPv6 and IPv4 services can be defined as an intermediate component between a native IPv4 host and a native IPv6 host to enable direct communication between them without requiring any modifications to the hosts. But if the host is a mobile node, triangle routing problem occurs, since Mobile IPv4 allows mobile node to roam transparently in any network. In this situation, mobile node must notify transfer information to its own home agent and correspondent node in IPv6 network. But current NAT-PT does not permit mobility header translation. Therefore, NAT-PT does not support efficient communication between Mobile IPv4 and Mobile IPv6. In this paper, we propose a mobility header conversion mechanism to resolve the triangle routing problem between Mobile IPv4 and Mobile IPv6.

I. 서 론

IPv4^[1]의 주소부족 문제를 해결하기 위해 IPv6^[2]가 개발되었다. IPv6는 무한의 주소공간과 QoS(Quality of Service) 제공 그리고 이동성지원 기능을 가지고 있어서 차세대 네트워크 프로토콜로서

각광을 받고 있다. 따라서 IPv4에서 IPv6로 대체하기 위해서는 IPv6의 도입시점부터 상당기간 IPv4와 공존해야하며 그 기간 동안 IPv4와 IPv6 간 통신 방안이 필요하다. 이에 IETF v6ops WG (IPv6 Operations Working Group)에서는 크게 세 가지 IPv4/IPv6 통신을 위한 메커니즘을 제안하였다. 첫

※ This work was supported by MIC and ITRC Project.

* 경희대학교 컴퓨터공학과 박사과정(dskim@khu.ac.kr), ** 경희대학교 전자정보대학 교수(cshong@khu.ac.kr)
논문번호 : KICS2006-08-349, 접수일자 : 2006년 8월 17일 최종논문접수일자 : 2007년 1월 2일

번째는 IPv4 스택과 IPv6 스택을 가지고 있는 이중 스택^[3]을 제안하였고 두 번째는 IPv4와 IPv6간의 터널을 통한 기법들^{[4][5]}을 제안 하였다. 마지막으로 IPv4와 IPv6 프로토콜을 변환할 수 있는 변환기^[6]를 제안 하였다. 이와 같이 유선 네트워크 환경에서 IPv4와 IPv6 간 통신을 하기 위해 여러 가지 메커니즘들이 제안되었고 일부 메커니즘들은 인터넷 표준이 되었다. 하지만 위와 같은 메커니즘들은 모바일 네트워크 환경을 고려하지 않은 메커니즘들이다. 따라서 모바일 환경을 지원하는 Mobile IPv4^[1]와 Mobile IPv6^{[7][9][12][13][14]}간의 효율적인 통신을 지원 할 수 있는 메커니즘들이 필요하다. 본 논문의 구성은 다음과 같다. 2장에서는 현재 IPv4와 IPv6 간의 변환 메커니즘들을 소개한다. 3장에서는 모바일 네트워크 환경에서의 변환 메커니즘의 필요성에 대하여 기술하고 본 논문에서 제안하는 Mobile IPv4와 Mobile IPv6의 Mobility 헤더 변환 알고리즘을 통해 Mobile IPv4 네트워크의 모바일 노드와 IPv6 네트워크의 호스트 간 통신 방안을 설명한다. 4장에서는 실험 결과를 보여주고 마지막 5장에서는 결론 및 향후연구 계획으로 마무리 한다.

II. 관련 연구

IPv6 도입으로 인해 네트워크의 형태는 그림 1과 같은 중간단계를 상단기간 유지할 것으로 예상된다. 그림 1과 같이 IPv4 네트워크와 IPv6네트워크 간에는 Translator^[6]를 사용하여 통신을 하고 IPv6 네트워크 사이에 IPv4 네트워크가 존재할 경우 두 IPv6 네트워크는 Tunneling^{[4][5]}을 통하여 통신을 할 수 있다. 그리고 IPv4 네트워크와 이중 스택을 지원하는 네트워크와는 IPv4로, IPv6 네트워크와는 IPv6로 통신을 할 수 있다. 따라서 IPv4와 IPv6의 변환 메커니즘은 크게 다음과 같이 분류 할 수 있다.

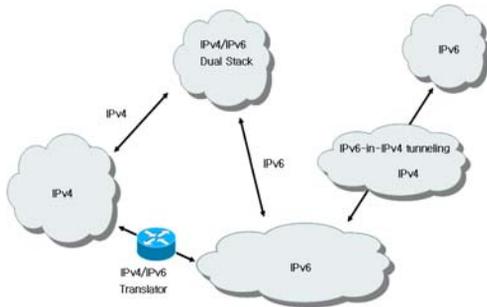


그림 1. IPv6 도입 시 예상되는 네트워크 구조

- ① Dual Stack
- ② Tunneling
- ③ Translator

2.1 Dual Stack

Dual Stack^[2]은 단말이 IPv4 프로토콜과 IPv6 프로토콜을 가지고 통신하는 기법이다. 따라서 IPv4와 IPv4 프로토콜을 사용하여 통신을 하고, IPv6와는 IPv6 프로토콜을 사용하여 통신을 한다. Dual Stack의 구조는 그림 2와 같다.

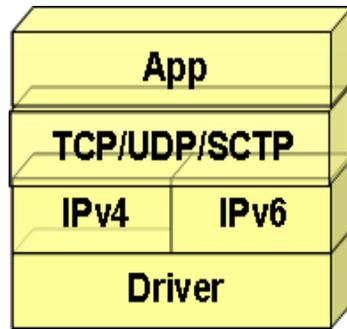


그림 2. Dual Stack의 구조

2.2 Tunneling

Tunneling에는 크게 두 가지 기법으로 분류할 수 있다.

2.2.1 Automatic Tunneling^[4]

이 기법은 IPv6 네트워크에 있는 Dual Stack 호스트가 IPv4 네트워크를 통과하여 다른 IPv6 네트워크에 있는 Dual Stack 호스트와 통신할 때 사용한다. 통신을 하고자 하는 호스트는 발신지와 목적지 주소를 IPv6 compatible 주소 타입으로 구성하여 IPv6 패킷을 만들어 IPv6와 IPv4 경계에 있는 라우터에게 보낸다. 라우터는 IPv6 패킷의 발신지와 목적지의 주소 중 IPv4 주소를 뽑아내어 IPv4 패킷을 만들어 터널링 하는 기법이다.

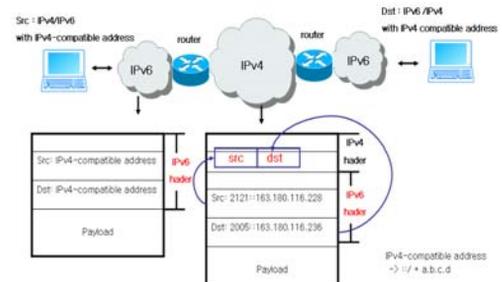


그림 3. Automatic Tunneling

2.2.2 Configured Tunneling^[5]

이 기법은 IPv6 네트워크에 있는 IPv6 호스트가 IPv4 네트워크를 통과하여 다른 IPv6 네트워크에 있는 IPv6 호스트와 통신할 때 사용한다. 통신을 하고자 하는 호스트는 IPv6 패킷을 만들어 IPv6와 IPv4 경계에 존재하는 라우터에게 보내면 라우터는 IPv4망을 통과하기 위하여 IPv4 패킷을 만들고 IPv6 패킷은 캡슐링하여 IPv4망을 통과하는 기법이다.

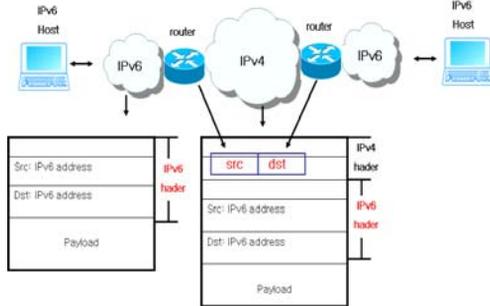


그림 4. Configured Tunneling

2.3 Translator

이 기법은 IPv4 호스트와 IPv6 호스트 간 통신을 위한 기법이다. IPv4 호스트에서 IPv6 호스트로의 통신에서는 IPv4 패킷을 경계 라우터가 IPv6 패킷으로 변환하고 IPv6 호스트에서 IPv4 호스트로의 통신에서는 경계 라우터가 IPv6 패킷을 IPv4로 변환하여 통신을 가능하게 만드는 기법이다. 헤더 변환은 그림 6과 같이 SIIT 알고리즘^[8]을 사용하여 변환하고 각 연결마다 임시로 할당할 IP 주소는 경계 라우터의 address pool^[6]에서 할당한다. Translator의 NAT-PT (Network Address Translation - Protocol Translation)^[6]의 구조는 그림 5와 같다.

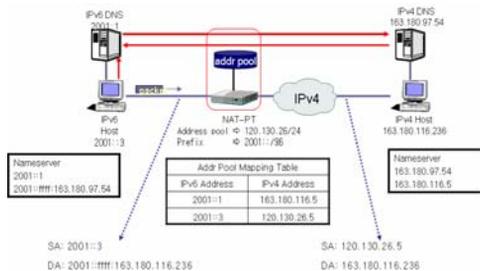


그림 5. NAT-PT의 구조

동작과정은 다음과 같다.

- ① IPv6 호스트가 IPv4 호스트의 주소를 알기위 해서 query 메시지를 보낸다.
- ② IPv4 호스트의 naming을 관리하는 DNS4 서버는 query 메시지에서 목적지 호스트의 IPv4 주소를 응답메시지에 포함하여 전달한다.
- ③ IPv6 호스트는 IPv6 패킷을 만들어 NAT-PT로 보낸다. 이때 목적지 주소는 IPv4-mapped 주소를 사용한다. (예) 2001::ffff:163.180.116.236
- ④ 패킷을 받은 NAT-PT는 IPv6 패킷의 목적지 주소에서 하위 32bit 필드의 IPv4 주소를 뽑아서 IPv4 패킷을 만든다. 이때 NAT-PT의 address-pool에서 source address를 임의의 IPv4주소로 할당하고 mapping table에 IPv6 source address 와 할당한 IPv4 address를 mapping해서 기록해둔다.

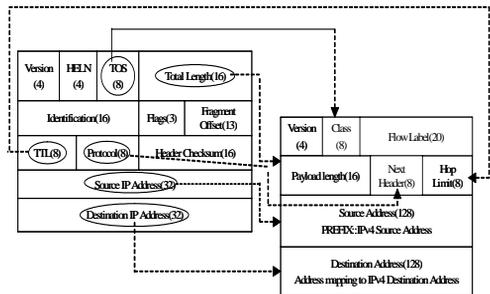


그림 6. SIIT 알고리즘(RFC 2765)

알고리즘은 다음과 같다

- Version field: 6 삽입
- Class field: IPv4 ToS 필드 값 복사
- Flow Label field: IPv4와 관련된 필드 없으므로 0을 삽입
- Payload field: IPv4 헤더의 Total length 필드 값 복사
- Next Header field: IPv4 헤더의 Protocol 필드 값 복사
- Hop Limit field: IPv4 헤더의 TTL 필드 값 복사
- Source Address field: IPv4-compatible 주소 타입 삽입
- Destination Address field: NAT-PT의 mapp-

ing 테이블에 있는 IPv6 호스트의 주소 값 삽입

IPv6 헤더를 IPv4 헤더로 변환 하는 알고리즘은 위의 IPv4 헤더를 IPv6 헤더로 변환 하는 알고리즘의 역과정이다.

III. 제안 사항

위와 같이 유선 환경에서 IPv4와 IPv6 변환 메커니즘들이 제안되어 통신이 가능하지만 모바일 네트워크 환경에서는 위와 같은 메커니즘들을 적용할 수 없다. 그 이유는 모바일 환경에서는 유선 환경과 다르게 단말들이 이동하기 때문이다. 무선 단말이 이동할 경우 새로운 IP를 부여받고 통신하고자 하는 단말에게 이동한 위치를 통보를 해 주어야 한다. 하지만 Mobile IPv4와 Mobile IPv6 간의 변환 메커니즘이 없어 단말이 이동시 다른 IP망에 있는 호스트에게 이동에 대한 통보를 할 수 없게 되어 효율적인 통신을 할 수가 없다. 따라서 본 논문에서는 다양한 모바일 환경 중에서 Mobile IPv4 네트워크의 모바일 노드와 IPv6 네트워크의 호스트 간 통신 방안을 제안 하였다.

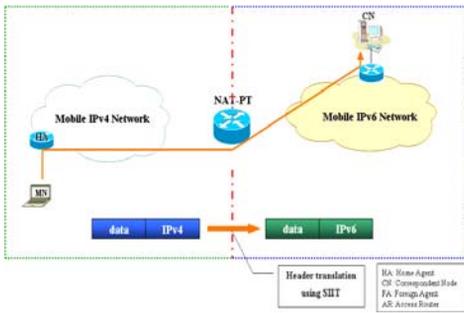


그림 7. Mobile IPv4 네트워크의 모바일 노드와 IPv6 네트워크의 호스트간 통신

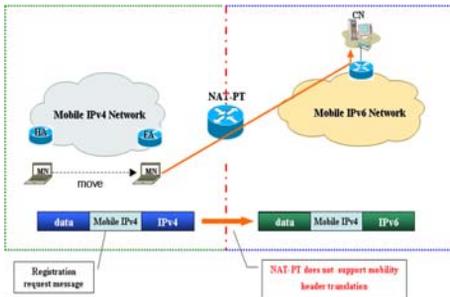


그림 8. 모바일 노드의 이동시 등록과정 문제

그림 7에서 모바일 노드가 자신의 홈 네트워크에 있을 경우 IPv6 네트워크에 있는 IPv6 호스트들과의 통신은 NAT-PT의 헤더 변환 메커니즘을 통해 통신을 할 수 있다. 하지만 그림 8에서 볼 수 있듯이 모바일 노드가 외부 망으로 이동시 Registration request^{[10][11]} 메시지를 IPv6 네트워크에 있는 호스트에게 보내지만 IPv6 호스트는 Registration request 메시지에 포함된 정보를 알 수가 없다. 결과적으로 IPv6 호스트는 모바일 노드가 이동해서 새롭게 부여받은 CoA(Care of Address)를 알 수가 없다. 따라서 그림 9와 같이 삼각형 라우팅 문제를 발생 시킨다.

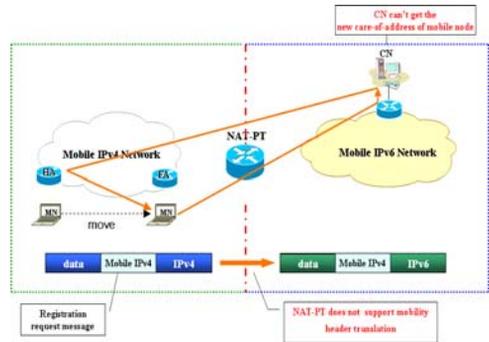


그림 9. 삼각형 라우팅 문제

위와 같은 삼각형 라우팅 문제를 해결하기 위해서는 NAT-PT에서 IPv4/IPv6 헤더 변환뿐만 아니라 Mobile IPv4/Mobile IPv6 헤더 변환도 반드시 지원해야 한다.

3.1 Mobile IPv4/Mobile IPv6 Mobility 헤더

그림 10은 Mobile IPv4 헤더^[2]와 Mobile IPv6 헤더^[3]의 포맷이다.

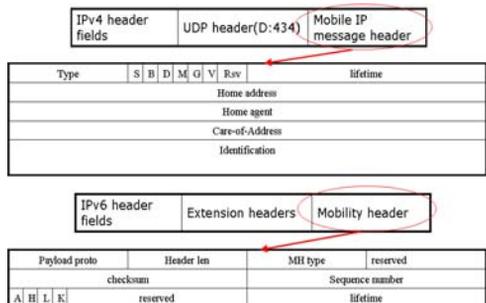


그림 10. Mobile IPv4 헤더와 Mobile IPv6 헤더

Mobile IPv4의 헤더는 8bits의 Type 필드, 6bits의 플래그와 16bits의 lifetime 필드, 32bits의 Home address, Home Agent, Care-of-Address 그리고 64bits의 identification으로 구성되어 있다. 각 필드의 의미를 살펴보면 다음과 같다.

- Type: 1(registration request를 의미)
- S flag: Home Agent에서 이전의 바인딩정보를 유지할 것을 요구
- B flag: 홈 네트워크로 전송되는 브로드캐스트 패킷을 HA가 CoA로 터널링 해줄 것을 요구
- D flag: Decapsulation
- M flag: Minimal encapsulation
- G flag: GRE encapsulation
- V flag: Van Hacobson header compression
- Home Address: 모바일 호스트의 고유 주소
- Home Agent: 모바일 호스트의 Home Agent 주소
- Care-of-Address: 외부망의로부터 부여 받은 주소
- Identification: 등록 시 reply 공격을 막기 위한 64bits 숫자

이와는 다르게 Mobile IPv6 헤더는 8bits의 Payload proto 필드, 8bits의 header len 필드, 8bits의 MH Type 필드, 16bits의 checksum 필드, 16bits의 sequence number 필드, 4bits의 플래그 그리고 16bits의 lifetime 필드로 구성되어 있다. 각 필드의 의미를 살펴보면 다음과 같다.

- Payload proto: IPv6의 next header 필드와 동일한 의미 (Mobility header: 59)
- header len: 헤더의 길이
- MH Type: Binding Update(5), Binding Acknowledgment(6)
- checksum: Mobility header에 대한 체크섬
- sequence number: 모바일 호스트가 여러 CN에게 보내는 Binding Update의 구분 값
- A flag: Binding Update에 대한 Acknowledgment 요구
- H flag: Home Agent로의 Binding Update 전송
- L flag: link-local address interface identifier와 home address가 같음
- K flag: Home Agent와 모바일 호스트간의

security association 요구

이와 같이 Mobile IPv4와 Mobile IPv6의 헤더의 위치 및 필드가 전혀 다르다. 따라서 IPv4 네트워크의 모바일 노드가 IPv6 네트워크의 호스트들과 통신을 하기 위해서는 NAT-PT에서 사용하는 IPv4와 IPv6에서 헤더 변환 알고리즘인 SIIT(Stateless IP/ICMP Translation Algorithm)[5]와 같은 Mobility 헤더 변환 알고리즘이 반드시 필요하다.

3.2 Mobility 헤더 변환 알고리즘

Mobility 헤더 변환은 Binding Update(Mobile IPv4에서는 Registration Request)와 Binding Acknowledgment(Mobile IPv4에서는 Registration Reply) 변환이 필요하다.

3.2.1 Binding Update 변환 알고리즘

IPv4 네트워크의 무선 단말은 이동 후 자신의 홈 에이전트와 CN(Correspondent Node)에게 새로운 CoA(Care of Address)를 포함한 Registration Request 메시지를 보내 등록을 해야 한다. 하지만 Mobile IPv6에서 사용하는 등록 메시지는 Binding Update 메시지이기 때문에 Mobile IPv4 Registration request 메시지를 Mobile IPv6 Binding Update 메시지로 변환 해야만 IPv6 네트워크의 호스트들은 IPv4 네트워크의 모바일 노드의 이동 위치를 알 수 가 있다.

Mobile IPv4의 Registration Request를 Mobile IPv6 Binding Update 메시지로 변환하는 알고리즘은 그림11과 같다.

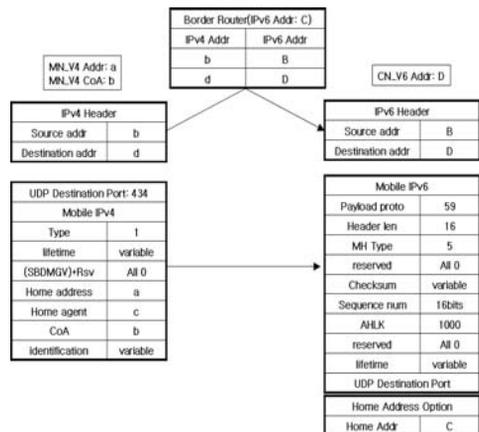


그림 11. Binding Update 변환 알고리즘

동작 과정은 아래의 순서와 같다.

- ① IPv4/IPv6 헤더 변환 : SIIT 알고리즘을 사용하여 변환
- ② Mobile IPv4/Mobile IPv6 헤더 변환
 - payload proto: Mobility Header를 의미하는 59 삽입
 - header len: 128bits ÷ 8bits = 16 삽입
 - MH Type: Binding Update 메시지를 의미하는 5 삽입
 - reserved: 0 삽입
 - checksum: 계산 후 값 삽입
 - sequence number: NAT-PT로부터 할당 받은 값 삽입
 - {AHLK}: 확인응답을 요구하는 A bits만 설정
 - lifetime: Mobile IPv4의 lifetime 값으로 삽입

위의 반대로 Binding Update 메시지를 Registration Request 메시지로 변환 하는 알고리즘은 위의 그림 11 알고리즘의 역과정이다.

3.2.2 Binding Acknowledgment 변환 알고리즘

Mobile IPv6의 Binding Acknowledgment 메시지를 Mobile IPv4 Registration Reply 메시지로 변환 하는 알고리즘은 그림 12와 같다.

- ① IPv4/IPv6 헤더 변환 : SIIT 알고리즘을 사용하여 변환
- ② Mobile IPv4/Mobile IPv6 헤더 변환

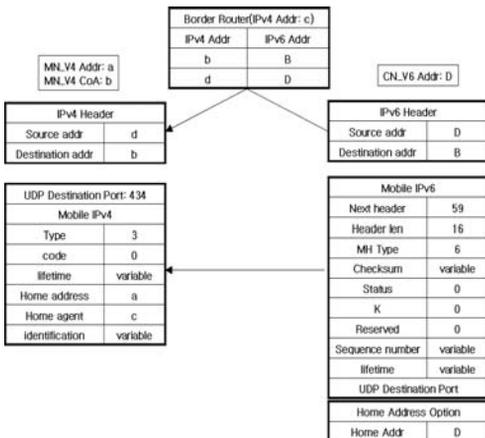


그림 12. Binding Acknowledgment 변환 알고리즘

동작 과정은 아래의 순서와 같다.

- UDP destination port: Mobile Header를 의미하는 434 삽입
- Type: Registration Reply 메시지를 의미하는 3을 삽입
- Code: Registration accepted를 의미하는 0을 삽입
- lifetime: Mobile IPv6의 lifetime 값으로 삽입
- Home address: BR에 저장되어 있던 자신의 IPv4 주소 삽입
- Home Agent: BR의 IPv4 주소를 삽입
- Identification: BR로부터 할당 받은 값 삽입

위의 반대로 Registration Request 메시지를 Binding Acknowledgment 메시지로 변환 하는 알고리즘은 위의 그림 12 알고리즘의 역과정이다.

3.3 등록 과정

등록 과정은 그림 13과 같다. 모바일 노드가 자신의 홈 망에서 외부 망으로 이동시 모바일 호스트는 새로운 CoA를 획득하고 CoA 정보가 포함된 Registration Request 메시지를 보낸다. 이때 NAT-PT는 받은 메시지를 Binding Update 변환 알고리즘을 사용하여 Mobile IPv6 Binding Update 메시지를 생성해서 CN에게 보낸다. CN은 이에 대한 응답으로 Binding Acknowledgment 메시지를 보내고 NAT-PT는 이 메시지를 Binding Acknowledgment 변환 알고리즘을 사용하여 Mobile IPv4 Registration Reply 메시지를 생성해서 모바일 노드에게 보냄으로써 등록 과정을 끝낸다.

제안 알고리즘의 동작과정을 그림 13과 같고, 상세 동작 시나리오는 그림 14와 같다.

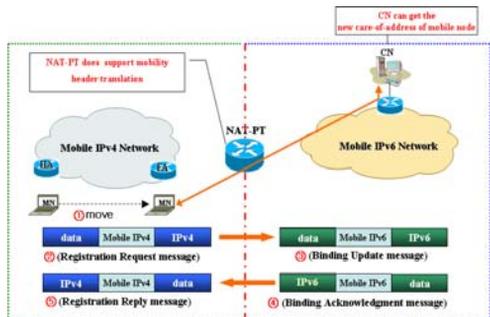


그림 13. Mobile IPv4 노드와 IPv6 CN간 등록과정

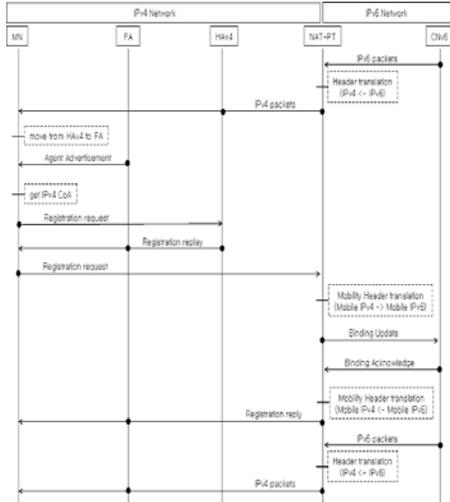


그림 14. 제안 알고리즘의 동작과정

동작 과정은 아래의 순서와 같다.

- ① 모바일 노드가 외부 망으로 이동
- ② 모바일 노드는 FA(Foreign Agent)로부터 Agent Advertisement 수신하고 새로운 CoA를 획득
- ③ 모바일 노드는 HA(Home Agent)에게 Registration request를 보내 새로운 CoA를 등록하고 이에 대한 응답으로 Registration replay를 수신
- ④ 모바일 노는 IPv6 네트워크의 CN에게 새로운 CoA를 포함한 Registration request를 보냄
- ⑤ NAT-PT는 모바일 노드가 보낸 Registration request 메시지를 Mobility 헤더 변환 알고리즘을 이용하여 Mobile IPv6 메시지인 Binding Update 메시지로 변환 하여 CN에게 전달
- ⑥ CN은 모바일 노드의 새로운 이동 위치를 확인 후 Binding Acknowledge를 모바일 노드에게 보냄
- ⑦ NAT-PT는 CN이 보낸 Binding Acknowledge 메시지를 Mobility 헤더 변환 알고리즘을 이용하여 Mobile IPv4 메시지인 Registration reply 메시지로 변환 하여 모바일 노드에게 전달

IV. 실험 및 결과

제안 알고리즘의 성능평가를 위하여 M/M/1 큐잉

모델을 사용하였다. End-to-End delay는 Propagation delay, Queuing delay 그리고 Transmission delay의 합이며 End-to-End delay를 구하기 위해 M/M/1 큐잉 모델[16]에서 정의한 아래 식을 사용하였다.

수식 1, 수식 2, 수식 3, 수식 4에서 사용한 기호의 의미는 다음과 같다.

- Service time(T_s): 노드에 들어온 패킷을 처리하는 시간
- Node utilization(ρ): 노드가 일을 하고 있는 비율이다. 예를 들어 node utilization이 1인 경우는 서버가 100% 일을 하고 있다는 의미로 도착한 패킷들은 서버에서 서비스를 제공할 수 없으며 큐에서 대기해야 한다.
- Transmission time(T_r): 노드에 들어온 패킷을 처리하고 전송하는 시간
- Propagation delay(P_d): 전파 지연 시간

$$Service\ time\ (T_s) : T_s = \frac{Packet\ length(bits)}{speed(bits/sec)} \quad (1)$$

$$Node\ utilization\ (\rho) : \rho = \lambda \times T_s\ (arrival\ packet\ rate) \quad (2)$$

$$Transmission\ time\ (T_r) : T_r = \frac{T_s}{1-\rho} \quad (3)$$

$$Propagation\ delay\ (P_d) : P_d = distance(m) \times (3 \times 10^8\ m/s) \quad (4)$$

수식 5와 수식 6에서 사용한 기호의 의미는 다음과 같다.

- TD: Total Delay
- DA_B: 노드 A에서 노드 B까지의 Propagation Delay
- PDA: 노드 A에서의 Processing Delay
- n: 네트워크에서의 라우터의 수

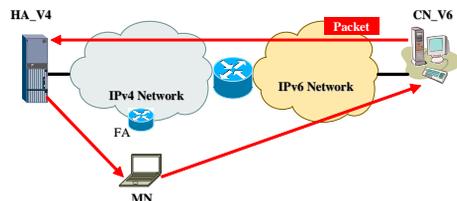


그림 15. SIIT 알고리즘만 적용된 네트워크에서의 Total Delay

$$TD = D_{CN_{NAT-PT}} + PD_{NAT-PT} + D_{NAT-PT_{HA}} + PD_{HA} + D_{HA_{MN}} = (2+n)PD + D_{CN_{MN}} \quad (5)$$

그림 15는 기존 NAT-PT가 적용된 네트워크에서의 Total Delay를 보여준다. 기존 NAT-PT가 적용된 네트워크에서는 모바일 노드가 이동시 IPv6의 CN은 모바일 노드의 이동 위치를 알 수가 없으므로 삼각형 라우팅을 통해 통신을 해야 한다.

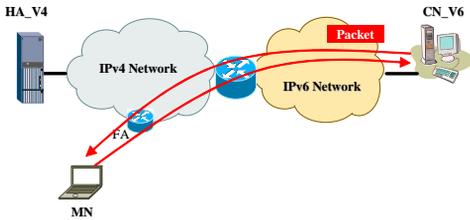


그림 16. 모바일 헤더 변환 알고리즘이 적용된 네트워크에서의 Total Delay

$$TD = D_{CN_{NAT-PT}} + PD_{NAT-PT} + D_{NAT-PT_{MN}} = (1+n)PD + D_{CN_{MN}} \quad (6)$$

그림 16은 본 논문에서 제안하는 모바일 헤더 변환 알고리즘이 적용된 네트워크에서의 Total Delay를 보여준다. SIIT 알고리즘뿐만 아니라 모바일 헤더 변환 알고리즘을 NAT-PT가 지원해 줌으로써 IPv6 CN은 모바일 노드의 이동 위치를 알 수 있다. 따라서 그림 15와 같은 삼각 라우팅을 발생하지 않는다.

실험은 IPv4 네트워크에서 홉 수를 증가 시킬 경우, IPv4 네트워크에 혼잡이 증가할 경우 그리고 IPv6 네트워크에 혼잡이 증가할 경우로 구분하여 End-to-End Delay를 기존 NAT-PT가 적용된 네트워크와 비교 하였다. 실험 매개 변수와 결과는 아래와 같다.

표 1. IPv4 네트워크의 홉 수의 증가를 포함한 매개 변수

Parameter	Simulation 1	
	IPv6 network	IPv4 network
Hop count	3	Increase value
Packet length	1000 bits	
Service time (each router)	0.12 sec	

Packet arrived rate (each router)	0.5 packet/sec
Transmission time (each router)	0.3 sec
Propagation delay	50,000m * (3 * 108m/s)

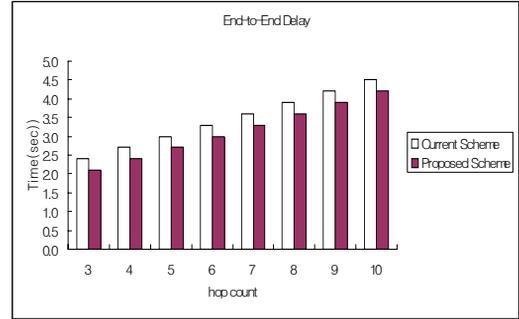


그림 17. IPv4 네트워크 홉 수를 증가한 실험 결과

그림 17의 실험결과에는 표 1의 매개 변수에서 다른 조건은 동일하게 두고 IPv4 네트워크의 홉 수를 증가 시키며 End-to-End Delay를 측정 한 결과이다. 모바일 노드가 자신의 홉 네트워크에서 점점 멀어져도 기존 NAT-PT가 적용된 네트워크 보다 더 적은 End-to-End Delay를 실험 결과가 보여준다.

표 2. IPv4 네트워크의 혼잡 증가를 포함한 매개 변수

Parameter	Simulation 2	
	IPv6 network	IPv4 network
Hop count	3	
Packet length	1000 bits	
Service time (each router)	0.12 sec	Increase value
Packet arrived rate (each router)	0.5 packet/sec	Increase value
Transmission time (each router)	0.3 sec	Increase value
Propagation delay	50,000m * (3 * 108m/s)	

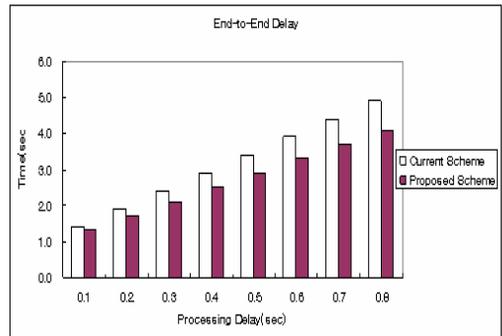


그림 18. IPv4 네트워크의 혼잡이 증가한 실험 결과

그림 18의 실험결과는 표 2의 매개 변수에서 다른 조건은 동일하게 두고 IPv4 네트워크의 혼잡을 증가 시키며 End-to-End Delay를 측정한 결과이다. IPv4 네트워크에 혼잡이 점점 더 증가해도 기존 NAT-PT가 적용된 네트워크 보다 더 적은 End-to-End Delay를 실험 결과가 보여준다.

표 3. IPv6 네트워크의 혼잡 증가를 포함한 매개 변수

Parameter	Simulation 3	
	IPv6 network	IPv4 network
Hop count	3	
Packet length	1000 bits	
Service time (each router)	Increase value	0.12 sec
Packet arrived rate (each router)	Increase value	0.5 packet/sec
Transmission time (each router)	Increase value	0.3 sec
Propagation delay	50,000m * (3 * 108m/s)	

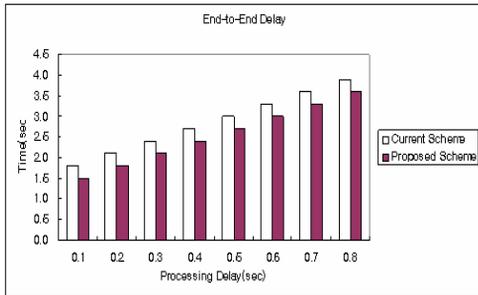


그림 19. IPv6 네트워크의 혼잡이 증가한 실험 결과

그림 19의 실험결과는 표 3의 매개 변수에서 다른 조건은 동일하게 두고 IPv6 네트워크의 혼잡을 증가 시키며 End-to-End Delay를 측정한 결과이다. IPv6 네트워크에 혼잡이 점점 더 증가해도 기존 NAT-PT가 적용된 네트워크 보다 더 적은 End-to-End Delay를 실험 결과가 보여준다.

위의 세 가지 실험에서 볼 수 있듯이 모바일 노드가 홈 네트워크로부터 멀어져도, IPv4 네트워크에 혼잡이 증가해도 그리고 IPv6 네트워크에 혼잡이 증가해도 기존 NAT-PT가 적용된 네트워크 보다 더 적은 End-to-End Delay를 보여주었다. 그 이유는 기존 NAT-PT가 적용된 네트워크는 IPv4/IPv6 헤더 변환만 지원하여 IPv6 네트워크의 호스트가 IPv4 네트워크의 모바일 노드의 이동 위치를 알 수

가 없으므로 삼각형 라우팅을 통해 통신하기 때문이다. 이와는 다르게 본 논문에서 제안하는 모바일 헤더 변환 알고리즘이 적용된 네트워크에서는 IPv6 네트워크의 호스트들이 IPv4 네트워크의 모바일 노드의 이동 위치를 알 수 있으므로 삼각형 라우팅 문제가 발생 하지 않는다.

따라서 기존 기법보다 더 적은 End-to-End Delay를 실험결과들을 통해 확인 할 수 있었다.

V. 결론

본 논문에서는 IPv4와 IPv6 간의 변환 기법에 대하여 살펴보고 무선 네트워크 환경에서 모바일 노드의 이동성으로 인해 발생하는 문제점을 기술하였다. Mobile IPv4를 사용하는 모바일 노드와 IPv6 네트워크에 있는 호스트간의 통신을 하기 위해서는 NAT-PT에서 IPv4/IPv6 헤더 변환 뿐만 아니라 Mobile IPv4/Mobile IPv6 간의 헤더 변환도 지원해야 하지만 기존 NAT-PT에서는 Mobile IPv4/Mobile IPv6 간 헤더 변환을 지원하지 않는다. 따라서 IPv4 네트워크에 있는 Home Agent가 터널링을 통해 외부 망에 있는 모바일 호스트로 패킷을 전달해줘야 한다. 하지만 이때 삼각형 라우팅 문제가 발생하고 비효율적인 라우팅이 발생하게 된다. 따라서 본 논문에서는 삼각형 라우팅 문제를 해결하기 위해 Mobile IPv4와 Mobile IPv6의 변환 알고리즘을 고안하여 모바일 노드와 CN과의 경로 최적화를 이룰 수 있는 방안을 제안하였다.

향후 과제로는 여러 가지 네트워크 상황에서의 Mobile IPv4와 Mobile IPv6 간의 변환 메커니즘들을 정의하고 구현을 통한 검증이 필요하다.

참고 문헌

- [1] C. perkins, Et al., "IP Mobility Support for IPv4", RFC 3220, January 2002
- [2] Deering, S. and R. Hinden., "Internet Protocol, Version 6 (IPv6)Specification", RFC 2460, December 1998.
- [3] K. Tsuchiya, H. Higuchi, Y. Atarashi., "Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS).", RFC 2767, February 2000.
- [4] C. Huitema, R. Austein, S. Satapati, R. van der Pol., "Evaluation of IPv6 Transition Mechanisms for Unmanaged Networks.", September 2004.

[5] D. Black., "Differentiated Services and Tunnels.", RFC 2983, October 2000.

[6] G.Tsirtsis, et al., "Network Address Translation - Protocol Translation (NAT-PT)", RFC 2766, February 2000.

[7] C. Perkins et al., "Mobility Support in IPv6", RFC 3775, June 2004.

[8] E. Nordmark., "Stateless IP/ICMP Translation Algorithm (SIIT).", RFC 2765, February 2000.

[9] I.Vivaldi, M. H. Habeabi, B. M. Ali, and V. Prakash., "Fast Handover Algorithm for Hierarchical Mobile IPv6 Macro-Mobility Management", In the Proceedings of The Asia-Pacific Conference on Communications, 3(270):630-634, September 2003.

[10] Z. G. Kan, J. Ma, J. Luo, and J. P. Hu., "Mobile IPv6 and Some Issues for QoS", Internet Society, 2001.

[11] T. Kato, R. Takechi, and H. Ono., "A Study on Mobile IPv6 Based Mobility Management Architecture", Fujitsu Sci. Tech., 37(1):65-71, June 2001.

[12] R. Koodli., "Fast Handovers for Mobile IPv6", Technical report, Mobile IP Working Group, October 2003.

[13] B. L. Ong and S. Hassan., "Mobile IPv6 Architectures and the QoS Issue on Handover Delay", In the Proceedings of National Seminar 2002, University Teknologi Mara, November 2003.

[14] H. Soliman, C. Castelluccia, K. E. Malki, and L. Bellier., "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)", RFC 4140, Network Working Group, August 2005.

[15] S. Zeadally and D. Mavatoor., "Mobile IPv6 Support For Highly Mobile Hosts", In the Proceedings of IASTED International Conference, pages 144-150, September 2003.

[16] Gunter Bolch et al., "Queueing networks and Markov chains", Wiley-Interscience, 2006.

김대선(Dae Sun Kim)



준회원

2002년 2월 경희대학교 전자계산공학과 졸업(학사)
2004년 2월 경희대학교 컴퓨터공학과 졸업(석사)
2004년 3월~현재 경희대학 컴퓨터공학과 박사과정
<관심분야> Mobile IP6, IPv6, NEMO, Sensor Networks

홍 충 선(Choong Seon Hong)

정회원

한국통신학회 논문지 제31 권 제10B호 참조