한국시뮬레이션학회 논문지
Vol. 15, No. 1, pp. 77-85 (2006. 3)

Ad Hoc 네트워크상에서 익명성을 보장하는 방법에 관한 연구

강승석[1†]

Provisioning Anonymous Communication in Ad Hoc Networks

Seung-Seok Kang

ABSTRACT

The cost of downloading content from the Internet may be costly for mobile device users using its 3G connection, because the 3G connection cost to download data from the Internet is a function of the amount of data downloaded. This paper introduces an approach in which mobile devices, called peers, form an ad hoc network and share their downloaded content with others. As an example, spectators may want to collect/share information about players and game records in a stadium. In an art gallery, visitors may want to retrieve some background information about the displayed work from the nearby ad hoc network. In an outdoor class, a teacher may download today's topic files from the Internet, and all students may share the content with minimal or no cost paid. This is possible if mobile device has both a 3G interface and a wireless LAN interface. If the peers want to improve privacy and discourage traffic analysis when sharing content, this paper describes a low-delay anonymous connection between the sending peer and the receiving peer using two additional peers. Simulation results show that the transmission time overhead of the anonymous connection may increase 50% or less as the number of peers increase or the peers are scattered over the larger area.

Key words : Anonymous Communication, ad hoc network, content sharing, cooperation of networks, communication cost reduction

요 약

무선기기가 인터넷에 저장된 자료를 3G 통신을 이용해 내려 받는 경우, 사용자가 부담하는 비용은 인터넷에서 내려 받은 자료의 양과 비례한다. 본 논문은 무선기기(peer 라 호칭함) 들이 애드 혹 네트워크를 구성하여 미리 내려 받은 자료를 다른 peer 들과 공유하는 방법을 소개한다. 예를 들어, 스포츠 경기장에서 선수나 경기에 관한 정보를 얻거나, 미술관에서 작품에 관한 내용을 공유하거나, 야외수업에서 선생님이 내려 받은 자료를 학생들이 무료로 그 내용을 공유할 수 있다. 이를 위해서는 무선기기가 3G 통신과 무선 LAN 통신이 가능해야 한다. 만약 무선기기 사용자가 다른 peer 와의 통신과정에서 익명성을 필요로 하고, 트래픽 분석(traffic analysis)을 어렵게 하고자 하는 경우, 본 논문에서는 애드 혹 네트워크 내에 있는 추가적인 두 peer를 이용하여 익명성을 보장하는 통신 방법을 기술한다. 애드 혹 네트워크에서 익명성을 제공하는 통신에 대한 모의실험을 수행한 결과, 통신시간에 대한 오버헤드가 익명성이 제공되지 않는 통신의 경우보다 50% 이하였으며, 참여하고 있는 peer의 수가 증가하거나 peer 들이 넓은 영역에 흩어져 있는 경우 오버헤드는 더 줄어들었다.

주요어 : 익명 통신, 애드 혹 네트워크, 콘텐츠 공유, 네트워크간의 협력, 통신비용 절감

1. 서 론

휴대용 무선기기를 이용하여 인터넷에 접속하는 서비스가 대중화되어 가고 있다. 휴대용 무선기기에 사용하는 통신방법으로는 3세대 통신 (3G) 서비스(WWAN)나 802.11과 같은 무선 근거리 통신망 (WLAN)을 이용할 수 있다. 3G 네트워크는 음성과 데이터 전송 시 최대 2.05 Mbps[1] 의 속도로 광역 서비스를 제공한다. 그러나 3G 통신을 이용해 인터넷에서 자료를 받는 비용은 내려 받은 자료의 양에 비례한다. 통신비용을 줄이기 위해서 내려 받은 자료를 다른 무선기기와 공유함으로 3G 통신

2006년 3월 3일 접수, 2006년 3월 13일 채택
[1] 서울여자대학교
주 저 자 : 강승석
교신저자 : 강승석
E-mail; msukang@swu.ac.kr

비용을 절감하거나[2,3], 내려 받고자 하는 자료의 일부분만 3G 통신을 이용해 내려 받고 내려 받은 자료를 비용이 부과되지 않는 채널을 이용해 서로 교환하여 전체 자료를 재구성하는 방법 등[4]이 제안되었다. 위의 연구를 위한 기본 가정은 각 휴대용 무선기기가 두개의 통신 채널을 가진다는 것인데, 하나는 WWAN을 접속하기 위한 유료의 3G 통신기능과 다른 하나는 WLAN을 이용하여 ad hoc 네트워크를 구성하기 위한 무료인 802.11이나 Bluetooth 등의 통신기능을 갖추어야 한다. 또한 ad hoc 네트워크에 참여한 모든 무선기기가 자료를 공유하거나 부분 자료를 가지고 전체 자료를 만들기 위해 서로 적극적으로 협력해야 한다는 것이다.

본 논문은 내려 받기 비용을 줄이기 위해 ad hoc 네트워크에서 익명성을 이용한 통신을 통해 무선기기끼리 자료를 공유하는 방법을 연구하였다. 먼저 peer 라고 하는 무선기기들이 자료를 공유하기 위하여 ad hoc 네트워크를 구성해야 한다. 각 peer는 이런 공유 기능을 도와주는 도움 서버와 일대일로 관계를 맺어야 한다. 도움 서버는 ISP나 CP(Content Provider) 혹은 인터넷 어느 곳에 위치해도 상관없다. 하나의 도움 서버는 여러 peer들과 관계를 맺을 수 있다. 때로는 하나의 도움 서버가 모든 peer들과 관계를 맺어 하나의 ad hoc 네트워크를 관리할 수도 있다. 관계를 맺은 도움 서버는 자신이 관리하는 peer들로부터 자신과 1-hop 떨어진 이웃 peer들의 정보를 얻는다. 또한 도움 서버는 자신의 peer들이 가지고 있는 파일 정보를 얻고 다른 도움 서버와 그 정보를 교환한다. 각 peer는 필요한 정보가 있는 경우 ad hoc 네트워크 내의 파일 관련 정보를 자신의 도움 서버를 통해서 존재 여부를 알 수 있다.

필요한 자료를 가지고 있는 peer와 그 자료를 원하는 peer가 정해지면, 두 peer가 통신을 하게 되는데, 이때 익명으로 하는 경우와 그렇지 않은 경우가 있을 수 있다. 송수신 peer 사이의 경로가 익명성을 가지게 되면 다른 peer들은 누가 송신 peer이고 누가 수신 peer인지 알기가 어렵게 된다. 유선 네트워크에서는 익명성과 관련된 여러 연구가 진행되고 있다. 예를 들면 Proxy를 기반으로 한 시스템인 Anonymizer[5]와 LPWA[6]는 requesting host와 source host 사이에서 작동한다. 수신자 익명성을 제공하는 시스템에 대한 연구는[7,8] 등이 있고, 송신자 익명성을 제공하는 시스템으로는 Mix[9], Onion[10,11], Crowds[12], Hordes[13], Tarzan[14] 등이 있다. 송수신자 상호 익명성을 제공하는 시스템으로 P5[15], APFS[16], Shortcut-Responding[17] 등이 있다. 무선 네트워크에서 위치에 대한 익명성을 제공하는 시스템에 대한 연구로는 mCrowds[18]와 [19] 등이 있다. 위의 무선 네트워크에서의 위치 익명성은 무선기기의 위치를 감추는데 중점을 두고 있지만, 본 논문은 무선 네트워크상에서 어떻게 peer들 끼리 자료를 공유하고 송신 peer와 수신 peer 사이에 익명성을 유지하면서 통신을 하는지에 중점을 두고 있다. 또한 각 peer는 유선 네트워크에 위치하는 도움 서버와 관계를 맺음으로 효율적인 ad hoc 네트워크 관리와 익명 채널을 관리할 수 있다. 익명 채널을 구축하는 책임은 자료를 요청하는 peer와 관계를 맺은 도움 서버에게 있다. 본 논문은 ad hoc 네트워크에서 익명성을 제공하는 방법과 익명성을 제공하는데 발생되는 오버헤드를 모의실험을 통해 기술하였다.

2. 익명 전송 채널 구축

본 논문에서는 ad hoc 네트워크에서 자료를 송신하는 peer와 자료를 수신하는 peer 사이에 프라이버시를 보장해주는 방법을 설명한다. 이 경우 수신 peer는 누가 실제 송신 peer 인지 알지 못하며 송신 peer도 누가 실제 수신 peer인지 알 수 없다. 또한 ad hoc 네트워크 내의 다른 peer들도 통신 분석 (Traffic Analysis)을 수행하기가 어려워진다. 통신 분석의 목적은 교환되는 자료의 내용을 분석하는 것이 아니라 누가 누구와 통신하는지를 분석하는 것이다. 비록 자료가 암호화 되어 있다고 하더라도 통신 분석은 통신하는 두 주체를 판별할 수 있다. 프라이버시를 위해서는 송신 peer와 수신 peer 사이에 익명 채널이 필요하다. 수신 peer의 도움 서버가 익명 채널 구축에 필요한 모든 일을 담당 한다. 유일하게 수신 peer의 도움 서버만 송신 peer와 수신 peer에 대한 정보를 알고 있으며, 다른 어떤 도움 서버도 이에 대한 정보를 알지 못한다. 또한 다른 어떤 peer들도 송수신 peer에 대한 정보를 알 수 없다.

2.1 ad hoc 네트워크 구성

익명성을 제공하는 ad hoc 네트워크를 구성하기 위해서는 유선 네트워크와 무선 ad hoc 네트워크간의 협력이 필요하다. 무선 ad hoc 네트워크를 구성하는 각 peer는 유선 네트워크에 위치한 도움 서버와 관계를 맺어야 한다. 하나의 도움 서버는 다수 혹은 ad hoc 네트워크 내의 전체 peer와 관계를 맺을 수도 있다. 도움 서버는 인터넷 상에 있는 ISP나 CP 혹은 기타 어느 곳에도 있을 수 있다. 무선기기는 ad hoc 네트워크에 참여하기 위하여 먼저 주변에 같은 목적을 가진 ad hoc 네트워크가 있는지 확인을

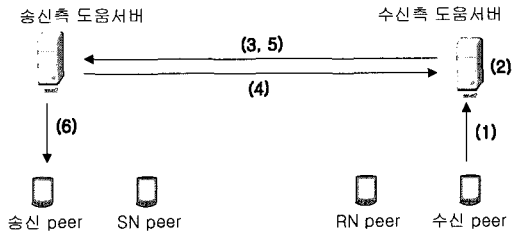
하여, 발견한 경우 그 네트워크에 합류하는 절차를 실행하고 발견하지 못하면 독자적인 네트워크를 구축하여 차후에 다른 무선기기가 합류할 수 있도록 한다. 이런 종류의 ad hoc 네트워크는 스포츠 경기장이나 전시장 또는 야외수업을 하는 경우 등 근방에 있는 여러 사용자들이 서로 자료를 공유하는 경우에 적합하다. 무선기기가 ad hoc 네트워크에 합류하는 세부 절차는 관련 논문들[2-4]에서 기술된 내용과 유사하다. 무선기기가 ad hoc 네트워크에 peer로 참여하여 인터넷에 위치한 서버에 등록을 하고 하나의 도움 서버와 관계를 맺게 되면, 그 peer는 자신이 저장하고 있는 파일 정보를 URL 형태[19]로 도움 서버에 전달한다. 또한 각 peer는 자신과 이웃한 peer를 발견하기 위해 HELLO 패킷을 간헐적으로 교환하여 이웃한 peer 정보 혹은 변경된 이웃 peer 정보를 도움 서버에게 알려준다. ad hoc 네트워크에 참여한 도움 서버들은 유선 네트워크상에서 서로 통신하면서 peer들의 요청을 처리한다.

2.2 익명 전송 채널 구축 절차

익명성을 제공하는 전송채널은 송신 peer와 수신 peer 사이에 구축된다. 통신 분석을 어렵게 하기 위해서는 두 peer 사이의 익명 전송 채널 구축 과정에서 추가적인 peer의 협조가 필요하다. <그림 1>에서는 익명 전송 채널의 구축 순서를 보여주고 있다. 수신 peer를 담당하는 도움 서버(수신측 서버)는 수신 peer와 1-hop 관계에 있는 peer들의 목록에서 하나의 peer(수신측 이웃 peer: RN peer)를 선택한다. 그리고 수신측 서버는 송신 peer를 담당하는 도움 서버(송신측 서버)에게 송신 peer와 이웃한 peer들 중 하나를 선택하여 선택된 peer(송신측 이웃 peer: SN peer)의 정보를 보내줄 것을 요청한다.

수신측 서버와 송신측 서버 사이의 통신은 wired network을 이용하며 통신 시 자료를 암호화하여 익명통신 구축 정보를 감춘다. 익명 전송 채널에서 자료의 전송은 송신 peer에서 시작하여 SN(Sender's Neighbor) peer, RN(Receiver's Neighbor) peer를 거쳐 수신 peer에게 전달된다. 이 경우, 모든 peer가 전송 범위 내에 있더라도, 패킷 하나를 송신 peer에서 수신 peer로 보내려면 세 번의 전송이 필요하다. 송수신 peer 사이의 세 경로 중에서 처음과 끝의 두 경로는 1-hop 이므로 이미 경로가 결정되어 있지만, SN peer 와 RN peer 사이의 경로는 알 수 없다. 그래서 두 peer 사이의 경로는 AODV[20]와 같은 ad hoc 라우팅 프로토콜을 이용해 찾을 수 있다.

익명의 수준과 송수신 peer 사이에 있는 중개 peer의



(1) 수신 peer가 도움서버에게 파일 F 를 요청
(2) 수신측 서버가 다른 서버와 Query를 통해 파일 F 위치 발견
(3) 송신측 서버에게 송신 peer의 이웃 peer 를 선택하도록 요청
(4) 송신측 서버는 SN peer 선택후 그 정보를 수신측서버에 전달
(5) 수신측 서버는 RN peer를 선택하고, 구성정보를 작성하여 송신측 서버에 전달
(6) 송신측 서버는 구성정보를 송신 peer에게 전달해 익명채널구성

그림 1. 익명 채널 구축 개략적인 과정

개수 사이에는 상충관계가 있다. SN peer나 RN peer 와 같은 중개 peer가 많을수록 익명도는 높아지지만, 반대로 최단거리 전송 경로를 선택하지 않기 때문에 자료 전송 시 오버헤드도 증가한다. 수신 peer 는 익명성의 수준을 선택할 수 있고, 수신측 서버는 이에 맞도록 중개 peer의 개수를 설정할 수 있다. 본 논문은 전송률을 고려하여 최소 개수의 중개 peer를 선택하였다. 자료의 전송율과 통신의 익명도는 중개 peer의 개수에도 영향을 받지만, 중개 peer를 선택하는 방식에도 영향을 받는다. Crowds[12]에서 사용한 무작위 중개 peer 선택 방식은 익명도를 향상시킨다. 본 논문은 수신 peer가 정한 익명도에 따라 수신측 서버가 적절한 수의 중개 peer를 선택할 수 있다. <그림 1> 에서는 송신 peer와 수신 peer로부터 1-hop 떨어진 이웃한 두 중개 peer를 선택하여 익명 채널을 구축하는 과정을 보여준다. 이런 1-hop 방식은 무작위 중개 peer 선택 방식보다 익명도는 떨어지지만 전송율은 높아진다. 다음 장의 모의실험에서 각 방식에서의 전송 성능 및 오버헤드를 비교하였다. 수신 peer와 송신 peer에 대한 정보를 유일하게 모두 알고 있는 수신측 서버는 익명 전송 채널을 구축하는 책임이 있다. 다른 어떤 도움 서버나 어떤 peer도 송수신 peer에 대한 정보를 모두 알지 못한다. 수신측 서버는 onion 라우팅[10]에서 사용하는 onion의 집합체와 유사한 익명 채널 구성 정보를 생성한다. <그림 2>는 익명 채널 구성정보에 대한 구조를 보여준다. 구성정보는 4개의 계층으로 된 자료구조이다. 각 계층은 똑같은 구조와 크기를 가진다. 각 계층은 목적지 peer 주소와 양방향 연결을 위한 3개의 암호화 키로 구성된다.

가장 밑에 있는 네 번째 계층을 제외한 나머지 세 계층은 네 peer 사이의 세 개의 가상 연결 정보를 가지고 있다.

목적지 주소	Padding 부분 암호키	Forward 자료 암호키	Backward 자료 암호키
SN peer	Sym Key P1	Sym Key E1	Null
RN peer	Sym Key P2	Sym Key E2	Sym Key E1
수신 peer	Sym Key P3	Sym Key E3	Sym Key E2
Null	Sym Key P4	Null	Sym Key E3
Padding			

그림 2. 익명 채널 구성 정보의 구조

다시 말해서, 가장 위에 있는 첫 번째 계층은 송신 peer와 SN peer 사이의 연결 정보를, 두 번째 계층은 SN peer 와 RN peer 사이의 연결 정보를, 세 번째 계층은 RN peer 와 수신 peer 사이의 연결 정보를 가지고 있다. 송신 peer 에서 수신 peer로 구성정보가 전달되면서 각 계층은 가장 위쪽의 첫 번째 계층부터 가장 아래쪽의 네 번째 계층 순서로 사용된다. 구성 정보는 수신측 서버에서 제작되어 송신측 서버를 거쳐 송신 peer에게 전달된다. 참여한 네 peer는 각각 구성정보를 받게 되면, 자신의 private 키를 이용하여 구성정보에 고정크기로 암호화된 목적지 주소와 암호화 키 등의 송수신 peer의 연결 정보를 얻는다. private 키를 사용하는 이유는 수신측 서버가 구성정보를 생성할 때 각 peer의 public키를 이용하여 암호화하였기 때문이다. 예를 들어 송신 peer가 구성정보를 통해 자신의 다음 목적지인 SN peer 주소와 전송자료를 암호화 하는 symmetric 키 정보를 알게 되면, 송신 peer는 자신이 읽은 첫 번째 계층의 정보를 삭제한 나머지 부분의 구성정보를 SN peer에게 전달한다. 송신 peer 입장에서는 자신이 읽은 계층 이외의 padding 부분은 수신측 서버가 구성 정보를 만들 때 관련된 각 peer의 public 키로 암호화하였기 때문에 익명 통신 정보를 더 이상 알 수 없게 된다. 각 peer가 다음 목적지 peer로 구성정보를 전송할 때 고정 크기의 자신을 위한 가장 위쪽 계층에 해당하는 부분을 삭제한 후, 주어진 세 개의 키 중에서 첫 번째 키로 구성정보의 padding부분에 적용한 후, 적용된 구성정보를 다음 목적지 peer에게 전송함으로 양방향 통신 경로를 설정할 수 있다. 세 개의 키 중에서 두 번째 키는 송신 peer에서 수신 peer로 forwarding 자료가 전달될 때 사용하고, 세 번째 키는 그 반대인 backward 자료 전송 시 암호화하는 용도로 사용된다. 세 번째 키는 송신 peer 에게는 필요 없다. SN peer가 구성정보를 받게 되면 송신 peer와 유사한 작업을 하고 새로 변경된 구성정보를 RN peer에게 전달함으로 가상의 통신 경로가 만들어진다. SN peer의 back- ward 키와 송신 peer의 forward 키는 같은 키를 사용한다. 각 키는 네 peer 중에서 논리적인 개념의 1-hop에서의 자료 전송 시 자료를 암호화 하는데 사용된다. 수신 peer가 구성정보를 RN peer로부터 받은 경우, 구성정보의 목적지 주소가 null이가 때문에 자신이 최종 목적지라는 것을 알게 된다. 위의 과정을 거치면 SN peer 와 RN peer의 도움을 받아 송신 peer에서 수신 peer에 이르는 양방향의 익명성 채널이 만들어진다.

익명 채널 구성 정보는 수신측 서버가 만든다. 수신측 서버는 먼저 구성정보의 가장 안쪽 계층을 만들고 padding의 크기를 결정한다. 가장 아래쪽 계층은 수신 peer의 public 키로 암호화 되고 padding 부분은 <그림 2>에 있는 P4 키로 암호화 된다. 계속해서, RN peer를 위해 두 번째 안쪽 계층을 만든다. 두 번째 계층은 RN peer의 public 키를 이용해서 암호화 하고 나머지인 padding 부분(수신 peer를 위한 가장 아래쪽 계층과 padding 부분)은 P3 키를 이용해 암호화 한다. 이 과정은 송신 peer를 위한 가장 위쪽 계층을 만들 때 까지 반복된다. 다시 말해서 가장 위쪽 계층은 송신 peer의 public 카로 암호화하고 나머지 부분(세 개의 계층과 padding)은 symmetric 키인 P1 으로 암호화 한다.

2.3 익명 채널을 통한 파일의 전송

익명 통신 채널이 만들어지면, 송신 peer는 수신 peer가 backward 방향으로 보낸 ACK 패킷을 받게 된다. 그러면 송신 peer는 자신의 가상 목적지인 SN peer에게 전송할 파일을 송신한다. 송신 peer와 SN peer는 1-hop의 거리에 있으므로 경로를 찾을 필요가 없다. 만약 둘 중 하나의 peer가 이동을 하여 연결이 끊어지게 되면, 송신 peer는 경로를 발견하는 ad hoc 라우팅 프로토콜을 이용하여 새로운 경로를 찾아야 한다. 중개 역할을 하는 SN peer나 RN peer를 찾을 수 없게 되면 송신 peer나 수신 peer가 이를 수신측 서버에게 알려야 한다. 수신측 서버는 다시 중개 역할을 할 peer들을 선택해서 새로운 구성정보를 만드는 작업을 반복한다.

SN peer와 RN peer 사이의 경로는 자료를 전송하기 전에 미리 경로를 알아야 한다. SN peer가 구성정보를 받으면 이를 통해 목적지인 RN peer에 대한 정보를 알 수 있으므로, 경로를 발견하기 위해 ad hoc 라우팅 프로토콜을 실행한다. SN peer는 송신 peer로부터 받은 자료를 ad hoc 라우팅 프로토콜을 이용해 발견한 경로로 RN peer에

게 자료를 전송한다. RN peer는 SN peer가 이 자료를 처음 전송한 송신 peer로 인식하게 된다. SN peer는 송신 peer로부터 받은 패킷의 payload는 변화시키지 않은 채 네트워크 레벨의 패킷 헤더만 바꾸어 재구성한 이후에 RN peer에게 전송한다. 같은 방법으로 RN peer는 수신된 자료를 목적지인 수신 peer에게 unicast로 전송한다. 수신 peer가 송신 peer에게 ACK와 같은 패킷을 보내야 할 필요가 있을 때 backward 방향의 경로를 이용한다. 만약 ad hoc 라우팅 프로토콜이 forward 경로를 찾으면서 동시에 backward 방향의 경로를 찾지 못한다면, RN peer는 SN peer까지의 경로를 찾아야 한다.

수신 peer가 원하는 파일의 수신이 완료되면, 수신된 파일에 결점이 있는지 검사 하여야 한다. 이를 위한 개략적인 과정이 <그림 3>에 있다. 자료를 전송하기 전에, 송신측 서버는 송신 peer가 저장하고 있는 파일의 무결성을 검사한다. 송신 peer는 송신측 서버가 제공한 초기 digest 값을 이용하여 MD5[21] 같은 message digest 알고리즘을 실행한다. 실행 후 계산된 최종 digest 값과 초기 digest 값을 수신측 서버에게 전달한다. 수신측 서버가 이를 수신 peer에게 전달하게 되면, 수신 peer는 전달받은 두 hash 값을 이용하여 수신된 파일의 무결성을 검사할 수 있다. 수신 peer는 받은 파일에 초기 digest 값을 적용하여 새로운 digest 값을 계산하고 이를 수신측 서버가 전달한 최종 digest와 비교하여 수신된 파일의 무결성 여부를 확인할 수 있다.

3. 모의실험 결과

익명성을 보장하는 경우 자료의 전송에는 오버헤드가 발생하는데, 이는 익명으로 자료 전달시 송신 peer와 수신 peer 사이에 더 많은 peer를 거쳐 가기 때문이다. 이번 장에서는 익명 통신을 이용하는 경우 발생하는 오버헤드를 ns2[22] 네트워크 시뮬레이터를 이용하여 모의실험을 하였다. 모의실험에서는 SN peer와 RN peer를 지정하고 송신 peer와 수신 peer 사이의 익명 채널을 설정하였다. 송신 peer는 500 Kbyte 크기의 파일을 TCP를 이용해 전송한다. 각 peer는 802.11 MAC 프로토콜을 이용해 250 미터 까지 전송가능하며 전송 속도는 2 Mbps 이다. 참여하는 peer의 수는 4에서 20 이다. 모든 peer는 가로 세로 600 미터인 평지에 위치해 있으며, 이보다 모의실험 영역의 크기가 크거나 작은 경우 따로 명시 하였다. 전송 완료 시간이 짧은 관계로 peer의 움직임 (Mobility)에 관한

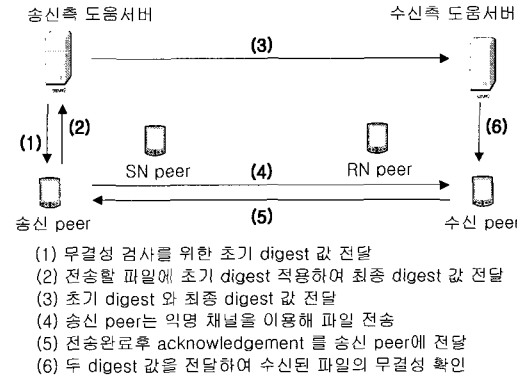


그림 3. 수신된 내용의 무결성 검사를 위한 개략적인 과정

특성은 고려하지 않았다. 실시된 모든 모의실험에서 ad hoc 네트워크는 연결되어 있다고 가정했다. 세 가지 TCP 패킷 크기를 사용하였으며 (256, 512, 1024 byte), 특별히 명시되지 않은 경우 모의실험에서는 1024 byte 크기의 패킷을 사용했다. 또한 모의실험에서는 세 가지 자료 전송 방법을 사용했다. 첫 번째는 "Direct" 방식으로 익명 채널을 이용하지 않은 경우이고, ad hoc 라우팅 프로토콜을 통한 경로로 자료를 전송한다. 다시 말해서, 모든 패킷들은 송신 peer에서 수신 peer까지 ad hoc 라우팅 알고리즘 (AODV)을 이용한 최단거리 경로를 통해 전송된다. 두 번째 방법은 "1-Hop" 방식으로, 중간에 있는 두 peer를 선택할 때, SN peer는 송신 peer의 1-hop peer 중에서 하나를 선택하고 RN peer는 수신 peer의 1-hop peer 중에서 하나를 선택한다. 세 번째 방법은 "Random" 방식으로 중간에 있는 두 peer인 SN peer와 RN peer를 1-hop peer가 아니라 참여한 peer 중에서 무작위로 선택한다. 오버헤드는 500 Kbyte 파일 전송을 완료하는데 까지 걸린 시간으로 측정하였다. 즉, 송신 peer가 첫 번째 자료 패킷을 전송한 후 500번째 TCP ACK 패킷을 받을 때까지의 시간이 총 전송 시간이 된다.

<그림 4>는 0부터 3까지 번호가 붙은 네 개의 peer로 구성된 다섯 개의 간단한 ad hoc 네트워크 구조를 보여주고 있다. 각 네트워크의 구조에서 0번 peer가 송신 peer가 되고 3번 peer가 최종 목적지인 수신 peer가 된다. 또한 1번 peer는 송신 peer의 다음 목적지가 되고 2번 peer는 1번 peer의 목적지가 된다. 최종적으로 2번 peer에서 수신 peer로 자료가 전송된다. Case A는 익명 채널을 이용하지 않고 송신 peer에서 수신 peer로 자료가 직접 전송되는 경우이다. Case B는 익명 채널을 이용하는 경우로 송신 peer에서 수신 peer로 자료 전송 시 3 hop이 필요하

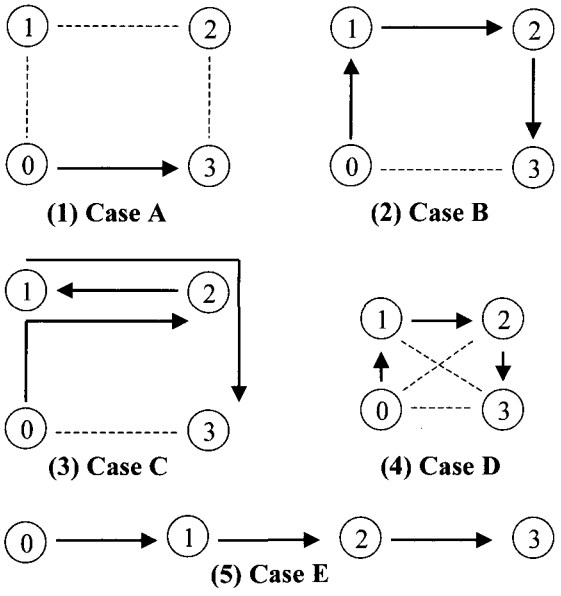


그림 4. 네 개의 peer로 구성된 다섯 가지 ad hoc 네트워크 구조

게 된다. 이 경우는 "1-Hop" 방식의 단순한 예라고 할 수 있다. Case C는 "Random" 방식의 한 예로서, 1번 peer가 0번 peer로부터 1 hop 보다 더 떨어져 있는 경우이다. 마찬가지로 2번 peer가 3번 peer와 2 hop 떨어져 있다. Case A, B, C 모두는 각 peer가 다른 두 peer와 직접 연결이 되어있지만, Case D의 경우는 네 peer 모두 직접 연결 되어있다. 다시 말해서 한 peer는 다른 모든 peer에게 직접 전송이 가능하다. Case E에서는 모든 peer가 일직선상에 위치한다. 이 경우에는 익명 채널을 통한 전송과 일반 전송 간의 오버헤드는 발생하지 않는다. 그 이유는 두 전송 방식 모두 같은 경로를 이용하기 때문이다.

<그림 5>는 0번 peer가 500개의 1024 byte TCP 패킷을 3번 peer에게 보내고 나서 각각에 대한 ACK 패킷을 받은 시간을 표시한 그림이다. Case A의 경우 전송하는데 걸린 총 시간이 10초 미만이었다. Case B, D, E는 20초 내외로 전송이 완료되었다. Case B는 Case D와 E보다 빨리 완료되었는데, 그 이유는 peer (0, 2) 와 (1, 3) 쌍이 서로 간섭을 하지 않기 때문이다. 반대로 Case E는 Case B와 D에 비해 조금 오래 걸렸는데, 그 이유는 1번과 2번 peer가 MAC level에서의 collision이 더 많았기 때문이다. Case C의 경우 전송 완료 시간이 30초를 초과하였다. 이 그림을 통해서 보면, 송신 peer와 수신 peer가 최단거리 (1-hop)에 있을 때 익명 통신의 오버헤드는 두 배가 된다. 두 peer 사이의 거리가 멀어질수록 오버헤드

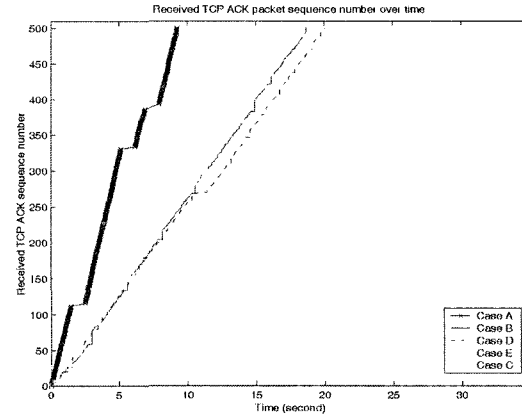


그림 5. 다섯 가지 기본적인 ad hoc 네트워크 구조별 자료 전송 완료 시간

는 줄어든다. 또한 "Random" 방식으로 전송하는 경우는 "1-Hop" 방식이나 "Direct" 방식보다 상당히 큰 오버헤드가 발생한다. "Random" 방식의 경우는 네트워크 내에 peer 수가 증가할수록 오버헤드가 더 커지게 된다.

<그림 6>은 여러 가지 다른 수의 peer가 ad hoc 네트워크에 참여했을 경우 송신 peer가 500번째 ACK 패킷을 받은 시간을 나타낸다. 위의 그래프를 포함한 모든 그래프에서의 수치들은, 무작위로 생성된 10개의 네트워크 구조를 가지고 각각의 구조에서 실험한 10번의 결과에 대한 평균이다. "Direct" 방식은 익명성을 제공하지 않으며 AODV 프로토콜에서 생성된 경로를 이용해 패킷들을 전

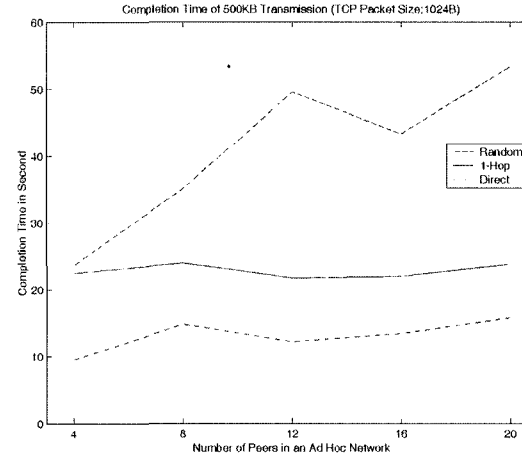


그림 6. peer의 개수별 전송 완료 시간

달한다. "1-Hop" 방식은 송신 peer와 수신 peer에서 1 hop 영역에 있는 두 peer를 선택하여 익명성을 제공하는 전송방식이다. 이 방식은 익명성의 정도가 낮으며 전송 시 지연되는 정도도 낮다. 세 번째 방식인 "Random"의 경우 SN peer와 RN peer를 임의로 선택하는데, 이 경우에는 익명성의 정도가 "1-Hop" 방식보다 높으며 또한 전송지연도 다른 두 방식에 비해서도 높다. 그래프를 통해 결과를 보면 참여한 peer의 수가 증가하는 경우 "1-Hop" 방식과 "Direct" 방식은 전송 시간이 일정하지만, "Random" 방식은 일반적으로 증가한다. 예외적으로 어느 위치에 있는 peer가 두 중간 peer로 선택되었느냐에 따라 전송 시간이 변화하기도 한다. 12 peer가 참여한 경우, 몇몇 ad hoc 네트워크 구조에서는 전송시간이 증가하여 그 평균값이 16 peer가 참여한 경우 보다 더 증가하였다. 8개의 peer로 이루어진 몇 개의 네트워크 구조에서는 "Direct" 방식의 경우, <그림 4>의 Case A와 Case B처럼 송신 peer와 수신 peer의 거리가 "1-Hop" 네트워크인 경우도 포함되어, <그림 6> 에서처럼 전송 평균 시간의 차이가 나기도 한다. 20개의 peer가 참여했을 때, "1-Hop" 방식의 경우 "Direct" 방식보다 50%의 시간이 전송하는데 더 필요했다.

<그림 7>은 12 peer가 참여한 경우로 TCP 패킷의 크기가 256 byte에서 1024 byte로 증가할 때 전송완료 되는 시간의 차이를 보여준다. 패킷의 크기가 256 byte인 경우, 송신 peer는 2000개의 패킷을 보내야 한다. 이 경우 "1-Hop" 은 "Direct" 방법보다 50% 정도의 시간이 더 걸렸다. 패킷의 크기가 4배 커진 1024 byte일 때 "1-Hop" 방식은 "Direct" 방식보다 78% 정도의 시간이 더 걸렸다. 패킷의 크기가 작을수록 전송완료 시간은 길어지지만 익명성에 의한 오버헤드는 감소한다.

<그림 8>은 1024 byte 크기의 TCP 패킷을 여러 크기의 모의실험 공간에서 전송할 때 완료된 시간을 측정하였다. 모의실험에 참여한 peer의 수를 12인 경우와 16인 경우로 하였다. 그래프 상에서 "Direct" 방법과 "1-Hop" 방식의 경우 밑에 있는 그래프는 12 peer가 참여한 경우이고, 위에 있는 그래프는 16 peer가 참여한 경우이다. "Random" 방법의 경우 공간의 크기가 400 미터와 600 미터인 경우 사이에서 교차하였다. 모든 peer가 다른 peer의 전송 영역에 들어가는 200 미터 공간에서의 실험 결과는 "1-Hop"에서의 경우 다른 공간의 크기에 비해 전송시간이 가장 오래 걸렸다. 그 이유는 한 순간에 하나의 peer만이 무선자원을 이용해 전송하기 때문이다. 다른 두 방식은 공간의 크기가 커질수록 전송시간이 증가하였다. 모

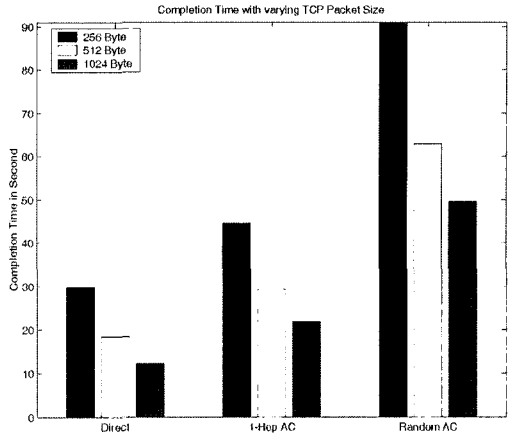


그림 7. 패킷 크기가 변할 경우 세 가지 방식별 전송 완료 시간

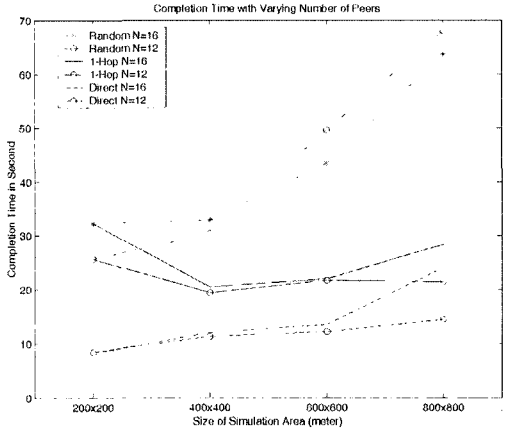


그림 8. 모의실험 영역이 변할 때 peer의 수에 따른 세 가지 방식별 전송완료 시간

든 peer가 1-hop의 전송범위 내에 있을 때, 익명성을 제공하는 통신의 경우 "Direct" 방식의 전송 시간에 비해 3-4배 정도 소요되었다. 일반적으로 모의실험 공간이 커질수록 전송완료 시간도 증가하는데, 그 이유는 송신 peer와 수신 peer 사이의 거리가 멀어지기 때문이다.

4. 결 론

본 논문은 ad hoc 네트워크에서 peer라고 부르는 통신기기들 중에서 두 peer가 익명으로 통신을 하고자 할 경우, 네트워크에 참여한 다른 두 peer들의 도움으로 익명

성을 유지하는 통신 방법을 소개하였다. 익명 채널에 참여한 송수신 peer에 관한 모든 정보는 오직 수신측 서버만 알고 있으며 어떤 다른 서버나 peer 들도 익명 채널에 대한 전체 정보를 알지 못한다. 본 논문에서 제안한 방법을 이용하는 경우 ad hoc 네트워크에서도 익명성을 보장하는 서비스를 이용할 수 있지만, 익명성으로 인해 발생하는 통신 성능의 저하는 불가피한 단점이다. 본 논문은 익명 통신을 이용하면서 발생되는 오버헤드를 익명성이 제공되지 않는 통신 방식과 비교하였다. 1-hop 익명성을 이용하는 경우 익명성을 이용하지 않는 방식과 매우 유사한 경향을 보였으며, 넓은 영역에 걸쳐 많은 peer들이 참여할수록 익명성을 이용함에 따라 발생하는 오버헤드는 감소하였다. 차후 연구 과제로는 ad hoc 네트워크에서 익명성을 제공하는데 부수적으로 발생하는 오버헤드를 줄이기 위한 다양한 연구가 필요하다.

참 고 문 헌

1. L. Garber, "Will 3G Really Be the Next Big Wireless Technology?", IEEE Computer, pp. 26. 32, Jan 2002.
2. S. Kang and M. Mutka, "Mobile Peer Membership Management to Support Multimedia Streaming", ICDCS Workshop on Mobile and Wireless Networks, pp. 770-775, May 2003.
3. S. Kang and M. Mutka, "Chumcast in Two-Tier Networks", Int'l Conference on Information Networking (ICOIN), pp. 523-532, Feb. 2004.
4. S. Kang and M. Mutka, "Efficient Mobile Access to Internet Data via a Wireless Peer-to-Peer Network" IEEE Int'l Conference on Pervasive Computing and Communications, pp. 197-205, March 2004.
5. E. Gabber, P. B. Gibbons, Y. Matias, and A. Mayer, "How to Make Personalized Web Browsing Simple, Secure, and Anonymous", Proc. of Conf. Financial Cryptography, pp. 17-31, Feb. 1997.
6. E. Gabber, P. B. Gibbons, D. M. Kristol, Y. Matias, and A. Mayer, "Consistent, yet anonymous, Web access with LPWA", Communications of the ACM, vol. 42, pp. 42-47, February 1999.
7. D. Chaum, "The Dining Cryptographers Problem: Unconditional sender and recipient untraceability", Journal of Cryptology, vol. 1, no. 1, pp. 65-75, 1988.
8. S. Dolev and R. Ostrovsky, "Efficient Anonymous Multicast and Reception", in Advances in Cryptography (CRYPTO97), August 1997.
9. D. Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", Communications of the ACM, vol. 24, pp. 84-88, February 1988.
10. M. Reed, P. Syberson, and D. Goldschlag, "Anonymous Connections and Onion Routing", IEEE Journal on Selected Areas in Communications, vol. 16, pp. 482-494, May 1998.
11. R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The Second-Generation Onion Router", in Proceedings of the 13th USENIX Security Symposium, pp. 303-320, August 2004.
12. M. K. Reiter and A. D. Rubin, "Crowds: Anonymity for Web Transactions", ACM Transactions on Information and System Security, vol. 1, pp. 66-92, November 1998.
13. C. Shields and B. N. Levine, "A Protocol for Anonymous Communication Over the Internet", in ACM Conference on Computer and Communications Security, pp. 33-42, November 2000.
14. M. J. Freedman and R. Morris, "Tarzan: A Peer-to-Peer Anonymizing Network Layer", in ACM Conference on Computer and Communications Security, pp. 193-206, November 2002.
15. R. Sharwood, B. Bhattacharjee, and A. Srinivasan, "P5: A Protocol for Scalable Anonymous Communications", in IEEE Symposium on Security and Privacy, pp. 53-65, May 2002.
16. V. Scarlata, B. N. Levine, and C. Shields, "Responder Anonymity and Anonymous Peer-to-Peer File Sharing", in Proceedings of ICNP 2001, pp. 272-280, November 2001.
17. L. Xiao, Z. Xu, and X. Zhang, "Low-cost and Reliable Mutual Anonymity Protocols in Peer-to-Peer Networks", IEEE Trans. on Parallel and Distributed Systems, vol. 14, pp. 829-840, September 2003.
18. C. Andersson, R. Lundin, and S. Fischer-Hubner, "m-Crowds: Anonymity on the Mobile Internet", in Proceedings of the 2nd IFIP Summer School, August 2003.
19. BitTorrent, "The BitTorrent file distribution system", http://bitconjurer.org/BitTorrent/
20. C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing", Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, Feb 1999.
21. R. Rivest, "The MD5 Message-Digest Algorithm", IEEE RFC 1321, April 1992.
22. ns2, The network simulator. http://www.isi.edu/nsnam/ns

강 승 석 (msukang@swu.ac.kr)

1992 고려대학교 이과대학 전산과학과 학사
1998 미시간주립대학교 공과대학 전산과학과 석사
2004 미시간주립대학교 공과대학 전산과학과 박사
2005 수원대학교 컴퓨터학과 전임강사
2006 서울여자대학교 컴퓨터학부 전임강사

관심분야 : ad hoc network, mobile computing, wireless communication, QoS, anonymous communication, multimedia communication