

IPv6 기반 Peer-to-Peer 확장 프로토콜*

박건우[†], 남민지[†], 조호식[†], 권태경[†], 최양희[†], 유태완[‡], 이승윤[‡]
 서울대학교 컴퓨터공학부[†], 한국전자통신연구원[‡]

Extended Peer-to-Peer Protocol based on IPv6*

Kunwoo Park[†], Minji Nam[†], Hosik Cho[†]
 Taekyoung Kwon[†], Yanghee Choi[†], Taewan You[‡], Seungyun Lee[‡]
 School of Computer Science and Engineering, Seoul National University[†]
 Electronics and Telecommunications Research Institute[‡]

요약

IPv4 주소공간의 고갈 문제를 해결하기 위해 많은 사람들은 IPv6의 도입을 추진하고 있다. 따라서 본격적인 보급을 눈앞에 두고 있는 IPv6는 6BONE과 같은 실험망에서 IPv6를 이용하여 개발된 여러 응용프로그램들로 그 성능 및 효율성을 테스트 받고 있다. IPv6 기반 P2P 확장 프로토콜 연구 및 응용 서비스 개발의 일환으로 Distributed Hash Table(DHT)을 이용한 P2P 오버레이 네트워크를 형성하는 톨을 소개한다. 이 톨은 인스턴트 메신저 기능을 합과 동시에 로그인한 사용자 모두를 대상으로 하는 파일 검색 및 공유 기능도 제공한다. 본 논문에서는 P2P 오버레이 네트워크의 구축 과정과 인스턴트 메신저의 친구 목록을 이용한 효과적인 파일 검색 기법을 증점적으로 설명한다.

1. 서론

현재 인터넷 체계는 32비트 주소 체계 기반인 IPv4(Internet Protocol version 4)을 사용하고 있다. 하지만 전세계적으로 인터넷이 보급되면서 사용 가능한 IP 주소가 빠르게 고갈됨에 따라 IP 주소를 획기적으로 늘리는 방법으로 128비트 기반의 IPv6 도입이 절실하게 되었다. 아직까지 대부분의 IPv6는 학교 및 연구소 중심으로 이루어진 실험망에서 여러 가지 테스트가 진행 중이고 실제 인터넷에는 소규모로 조금씩 도입되고 있다. 멀지 않은 미래에 IPv6가 전세계적으로 사용될 것에 대비하여 여러 기관에서는 IPv6를 사용하는 시험 어플리케이션을 개발하여 다각적인 방법으로 연구가 이루어지고 있다 [1].

이와 같은 상황에서 본 논문은 IPv6 환경에서 작동하는 P2P 확장 프로토콜의 연구와 해당 프로토콜을 실제로 사용하는 응용 프로그램을 소개한다. P2P 오버레이 네트워크를 형성하고 인스턴트 메신저 기능뿐만 아니라 사용자간의 파일 공유 기능 또한 제공한다. 정보의 검색은 전 노드에 걸친 DHT(Distributed Hash Table)의 효과적인 배치와 인스턴트 메신저의 친구 목록을 적극적으로 이용하여 정확하고 빠르게 이루어진다. 추후에 오버레이

멀티캐스트 망을 구축하여 다자간 회의, 비디오 스트리밍 서비스 등을 추가로 제공할 예정이다.

이후 본 논문의 구성은 다음과 같다. 2장에서는 기존의 오버레이 P2P 알고리즘에 대해서 논한다. 3장에서는 개발된 톨의 개요 및 작동 원리에 대해서 알아본다. 4장에서는 개발된 톨의 성능 분석 및 기대 효과를 추측해 보고 5장에서는 본 논문의 결론을 제시한다.

2. 기존에 연구된 알고리즘

Peer-to-Peer(P2P)기술은 HTTP와 같이 어플리케이션 레벨의 프로토콜로, 사용자 사이의 실시간 통신이나 자원분배 및 교환 등을 지원하는 기술로 정의되고 있다. 기존에 개발된 오버레이 P2P 알고리즘 중 대표적인 CHORD와 CAN에 대해 간단히 알아본다.

2.1 CHORD

Peer-to-Peer 어플리케이션에서의 기본적인 문제는 데이터 아이템을 저장하는 노드의 효율적인 배치이다. CHORD는 이러한 문제를 해결하기 위한 distributed lookup protocol이다. CHORD는 단지 하나의 오퍼레이션만을 제공하는데, 그것은 주어진 key에 대해서 노드를 매칭하는 것이다. CHORD는 노드의 join과 leave에

* 본 논문은 2004년도 한국전자통신연구원(ETRI)과 두뇌한국21, 국가지정연구실 프로젝트 지원을 받아 수행되었음

효율적으로 대처하며 시스템이 계속적으로 변화하는 과정에서도 주어진 오퍼레이션을 처리할 수 있다. 시스템 유지와 통신을 위한 비용은 노드의 수에 로그적으로 증가한다. 파일 공유 프로그램은 CHORD 위에서 손쉽게 구현할 수 있다. 파일 이름을 이용해서 노드와 파일정보를 매칭시킨다. 파일정보의 관리와 다운로드 서비스를 제공한다 [2].

2.2 CAN

CAN에서 저장 장소는 가상의 데카르트 좌표로 구성된 것으로 hash table과 유사하다. 전체 공간은 모든 노드가 개개의 독립된 chunk를 갖도록 분리되어 있으며 데이터는 이 공간 상의 한 포인트에 저장된다. 또한 데이터를 찾기 위해서는 포인트와 포인트 사이로 라우팅 할 수 있다. 포인트란 각 노드가 소유하는 독립된 구역이다. (key, value)의 pair로 되어 있는 데이터는 해쉬 함수를 사용하여 저장될 위치를 찾아 그 곳에 저장한다. 또한, 사용자는 찾고자 하는 (key, value)값을 해쉬함수를 사용하여 저장 위치를 알아내고 그곳에서 파일을 검색한다 [3].

3. 오버레이 Peer-to-Peer 알고리즘

이번 절에서는 기존의 P2P 오버레이 프로토콜과는 달리 본 연구진이 개발한 IPv6 상에서 작동하는 새로운 P2P 오버레이 프로토콜을 소개한다. 모든 노드는 처음으로 오버레이 네트워크에 참여하기 위해서는 key 노드에 join 메시지를 보내게 된다. 따라서 key 노드라는 것은 언제나 접속 가능하고, 모든 노드가 처음부터 그 주소를 알고 있는 특수한 노드로 가정한다. 성공적으로 오버레이 네트워크에 참여한 노드는 자신의 ID에 맞는 적절한 resource locator를 배당 받는다. Resource locator는 오버레이 상에서 자신의 좌우 노드의 주소뿐만 아니라 인스턴트 메시지 기능중의 하나인 친구 리스트에 대한 주소도 포함한다. 데이터의 위치를 찾는 명령이 내려지면 해당 데이터의 resource locator의 위치와 가장 가까운 노드에 명령이 전달되고 해당 노드는 데이터의 resource locator를 반환함으로써 데이터가 위치한 실제 노드의 주소를 알 수 있게 된다. 이 과정에서 친구 목록을 오버레이 네트워크를 형성한 링 상에서 일종의 지름길로 이용하기 때문에 효율적인 검색이 가능하다.

3.1 오버레이 노드

오버레이 노드는 Distributed Hash Table(이하 DHT)를 이용한다. 각각의 노드에 있는 DHT에는 오버레이 상의 데이터에 대한

resource locator가 적절히 들어있다(3.5 참고). 자신이 갖고 있는 데이터에 대한 검색 요청이 들어오면 해당 데이터의 위치를 가르쳐주는 resource locator를 반환하게 된다.

3.2 Resource Locator

Resource Locator는 해당 resource가 어느 IP에 어떠한 port를 사용하고 있는가 하는 정보를 담고 있다. 그림 1은 resource locator의 형태를 보여준다.

ID	IP	Port
----	----	------

그림 1

여기에서 resource는 ID로 구분되는데 사용자의 ID가 될 수도 있고 파일의 ID가 될 수도 있다. 사용자의 ID는 사용자의 E-mail address를, 파일의 ID는 파일 이름을 MD5 hash algorithm에 대입하여 얻을 수 있다 [4].

3.3 오버레이 네트워크 형성 과정

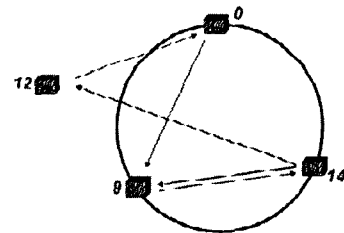


그림 2

오버레이 상에서 key node의 ID는 항상 0번으로 고정된다. 그림과 같이 0, 9, 14번 노드가 오버레이를 형성하고 있고 12번이 처음으로 join을 할 경우 12번은 key node에게 JOIN 메시지를 보낸다. Key node는 JOIN 메시지를 처리할 수 있는 적절한 노드인 14번까지 JOIN 메시지를 전달해 준다. 14번은 12번 노드에게 왼쪽과 오른쪽에 놓일 노드를 가르쳐 주고, 9번 노드에게 오른쪽 노드가 12번으로 바뀌었다는 사실을 전달한다. 또한 14번 자신의 왼쪽 노드를 12번 노드로 갱신함으로써 12번 노드는 성공적으로 오버레이 네트워크 상에 참여하게 된다.

3.4 라우팅

모든 노드는 오버레이 네트워크의 유지 및 복구를 위해서 2 hop 내에 있는 네 개의 노드에 대한 resource locator를 갖고 있다. 친구 목록을 유지하기 위해서 모든 친구에 대한 resource locator 또한 갖고 있다.

* 본 논문은 2004년도 한국전자통신연구원(ETRI)과 두뇌한국21, 국가지정연구실 프로젝트 지원을 받아 수행되었음

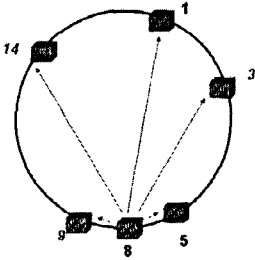


그림 3

그림 3과 같이 8번 노드의 친구로 14, 1, 3번이 등록되어 있고, 9, 5번은 8번의 왼쪽, 오른쪽에 위치했다고 가정한다. 만약 8번 노드에 16번 데이터를 찾아달라는 명령이 내려지면 이 메시지는 모든 노드를 돌면서 전달되는 것이 아니고 바로 14번 노드에게 전달된다. 즉 친구 리스트는 일종의 지름길 기능을 함으로써 오버레이 네트워크의 크기가 상당히 커지더라도 원하는 데이터를 몇 hop 내로 검색이 가능하도록 한다.

3.5 정보의 저장

오버레이 상에 노드가 join 또는 leave 하거나 데이터가 새로 생성, 삭제될 때마다 각각의 노드가 관리해야 할 resource locator가 달라져야 한다. 따라서 모든 노드는 자신의 'share' 디렉터리 내에 있는 파일들의 resource locator들을 STORE 메시지를 사용하여 주기적으로 내보낸다. STORE 메시지는 오버레이 상을 돌면서 해당 노드의 ID와 비교되면서 적절한 위치에 resource locator를 배치시킨다. 만약 일정 시간이 지났음에도 불구하고 특정 데이터에 대한 STORE 메시지가 오지 않는다면 해당 데이터는 삭제되었다고 생각하고 그 데이터의 resource locator는 지워지게 된다.

4. 성능분석과 기대효과

전체 n명의 사용자가 평균적으로 m명의 친구 목록을 유지하고 있다면 한번의 검색 명령당 $O(n/m)$ 개의 노드를 거친 후 검색 명령이 내려진다. 또한 오버레이 네트워크 상에서 데이터 파일의 전체 개수가 t 일 경우 각각의 노드는 t/n 개의 resource locator만 유지하면 파일 검색이 성공적으로 이루어질 수 있다. DHT를 이용하여 모든 노드가 동일하게 t/n 개의 resource locator를 유지하기 때문에 사용자가 많아지거나 공유되는 데이터 파일의 규모가 커지더라도 각 노드에 미치는 부담을 최소화 할 수 있다.

본 연구진에서 개발한 P2P 오버레이 네트워크를 이용한 인스턴트 메시지 및 친구목록을 이용한 파일 공유 시스템은 기존의

CHORD나 CAN에 비해 알고리즘이 상대적으로 간단하면서도 좋은 성능을 보여주었다. 또한 구현의 용이함으로 인해서 여러 방면에서 응용될 수 있을 것으로 보인다. 보다 실제 상황에 가까운 결과를 얻기 위하여 대규모 오버레이 네트워크를 구성하여 테스트하는 방법을 강구해야 할 것이다.

5. 결론

본 연구팀은 IPv6가 본격적으로 도입될 차세대 인터넷에서의 오버레이 네트워크를 이용한 peer-to-peer 응용과 multicast 응용을 염두에 두고 새로운 P2P 확장 프로토콜을 고안했다. 인스턴트 메시지 기능인 친구 리스트의 활용으로 많은 수의 노드에서도 효과적으로 파일 공유가 가능하다. 본 연구팀은 추후에 오버레이 멀티캐스트 응용을 통해 다자간 회의 및 비디오 스트리밍 서비스를 제공할 예정이다.

참고 문헌

- [1] 신명기, 김용진, "IPv6 도입을 위한 기술개발 동향", IPv6 포럼 코리아 기술문서 2000-002, <http://www.ipv6.or.kr>
- [2] Stoica, I.; Morris, R.; Liben-Nowell, D.; Karger, D.R.; Kaashoek, M.F.; Dabek, F.; Balakrishnan, H.; "Chord: a scalable peer-to-peer lookup protocol for Internet applications", Networking, IEEE/ACM Transactions on , Volume: 11 , Issue: 1 , Pages:17 - 32, Feb. 2003
- [3] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp and Scott Shenker, "A Scalable Content-Addressable Network", SIGCOMM, 2001
- [4] RFC 1321, "The MD5 Message-Digest Algorithm"

* 본 논문은 2004년도 한국전자통신연구원(ETRI)과 두뇌한국21, 국가지정연구실 프로젝트 지원을 받아 수행되었음