情 報 保 護 學 會 誌
第 15 卷 第 6 號, 2005. 12

DHT(Distributed Hash Table) 기반 P2P 오버레이 네트워크 보안 위협 분석

권 혁 찬*, 나 재 훈*, 장 종 수**

요 약

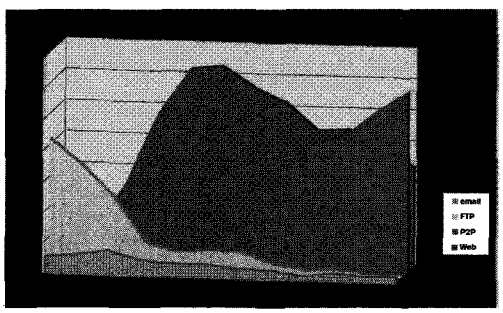
올 9월 NGN2005에서는 2003년을 기점으로 하여 인터넷에서 가장 많은 트래픽을 차지하는 서비스가 P2P(Peer-to-Peer) 서비스라는 통계를 발표하였다. 실제로, 현재 수많은 사용자가 P2P 서비스를 이용하고 있으며 다양한 P2P 응용 및 구조에 대한 연구가 다소 활발히 진행되고 있는 상황이다. 최근에는 P2P 파일공유 응용분야에서 DHT(Distributed Hash Table) 기반의 오버레이 네트워크를 활용하고자 하는 일부 연구도 진행되고 있다. 이와 관련하여 본 고 에서는 DHT 기반 P2P 오버레이 네트워크를 구축하기 위한 기존의 방식들을 소개하고, 이에 대한 보안 위협을 분석하고자 한다.

Ⅰ. 서 론

P2P 서비스하면 얼마 전까지만 해도 특히 저작권 문제 등이 대두 되면서 '문제아' 정도의 취급을 받은 것이 사실이다. 그러나 최근 들어서는 P2P를 하나의 통신을 위한 계층으로 바라보자는 시각도 등장할 정도로 인터넷에서 매우 대중화된 서비스가 되어 버렸다. 실제로 NGN2005에서는 2003년 이후로 인터넷에서 가장 많은 트래픽을 발생하는 서비스가 바로 P2P라는 통계를 발표하였다.[1] - [그림 1].

문제는 P2P 서비스의 부작용 - 저작권 문제, 보안 문제 등 - 을 최소화 하면서 안전하고 효율적인 방법으로 P2P를 사용하는 환경을 조성하는 것이다. P2P 서비스는 여러 가지 기준에 의해 분류가 가능하다. 응용에 따라 Instance Messaging, File Sharing, Distributed Computing 등의 응용으로 구분할 수 있다. 파일 공유 서비스의 경우, 자원검색 구조에 따라 Hybrid 구조, Distributed 구조로 분류할 수 있다. Hybrid 구조는 서버가 인덱스목록을 중앙 집중으로 관리하는 방식이다. 자원을 검색하고자 하는 피어는 중앙서버로 검색요청을 하면, 중앙서버는 요청한 자료의 검색결과를 보내준다. 이를 수신한 피어는 목록을 보고 직접 자원을 보유한 피어로 접속하여 자원을 요청하는 구조이다. 최근의 대부분의 파일공유를 위한 P2P 응용은 이러한 방식을 취하며 대표적으로 Napster[2]가 이런 서비스를 제공한다. Distributed 구조는 Gnutella[3]와 같이 서버가 존재하지 않고 브로드캐스트(Broadcast) 기반의 동작을 통해 자원검색을 수행하는 구조를 갖는다. 보통 검색의 효율성을 높이기 위해 브로드캐스트 방식의 다양한 변형들이 많이 제안되어 왔지만, 실제 검색과정에서 너무 많은 트래픽이 발생하는 문제가 있다.

이러한 배경에서 등장한 것이 DHT(Distributed Hash Table) 기반의 오버레이 네트워크 이다. DHT 기반의 오버레이 네트워크는 브로드캐스트 기반이 아닌 정형화된 검색 방식을 채택한다. 이것이 가능한 이유는 응용계층에서 관리하는 라우팅 테이블의 사용과 DHT 기반의 알고리즘 동작 때문이다. 따라서 기존의



[그림 1] 인터넷 트래픽 추세

* 한국전자통신연구원 정보보호연구단 P2P 보안연구팀 (hckwon@etri.re.kr, jhnah@etri.re.kr),
** 한국전자통신연구원 정보보호연구단 네트워크보안그룹 (jsjang@etri.re.kr)

Distributed 방식을 브로드캐스트 기반으로 동작하는 Unstructured Distributed 방식으로, DHT 기반의 오버레이 네트워크는 Structured Distributed 방식으로 분류하기도 한다.

본 고에서는 DHT 기반 P2P 오버레이 네트워크 구축을 위해 기존에 제안된 방식들에 대해 살펴보고, 이에 대한 보안 위협을 분석해 보고자 한다. 먼저 2장에서는 DHT 기반의 오버레이 네트워크 구축을 위한 방식들을 소개하고, 3장에서 DHT 기반의 오버레이 네트워크에서 예상되는 보안 위협을 분석한다. 마지막으로 4장에서 결론을 맺는다.

II. DHT 기반 오버레이 네트워크

DHT 기반 오버레이 네트워크의 구조를 그림 2에서 간략하게 보여주고 있다. DHT 기반 오버레이 네트워크에서는 파일의 위치정보를 분산시킨다. 그림 2에서 관리하는 정보인 (K,V) 쌍은 Key와 Value의 약자로서 Key는 검색하고자 하는 파일의 이름을 hashing 한 값이고, Value는 그 파일을 소유하고 있는 피어의 IP 주소이다. 그림 2에서는 K_1이라는 key의 정보를 찾는 과정을 보여주고 있으며 찾아가는 과정이 붉은 화살표로 표현되어 있다. DHT 기반 오버레이 네트워크는 P2P 오버레이 네트워크 형성에 구조적인 특성을 부여한 것으로 K_1정보 검색 메시지는 K_1과 각 노드의 라우팅 테이블의 내용에 따라 정해진 패스로 전달되며 이동한다. 대표적인 DHT 기반 오버레이 구축 방식으로 현재 Chord[4], Pastry[5], Tapestry[6], CAN[7]이 있다. 이 중 Chord, Pastry, Tapestry는 원형의 식별자 공간을 갖는다는 유사점이 있으며 CAN의 경우는 다차원의 공간을 갖는 다는 차이점이 있다. 본 절에서는 Chord와 CAN에 대한 구조를 분석하였다.

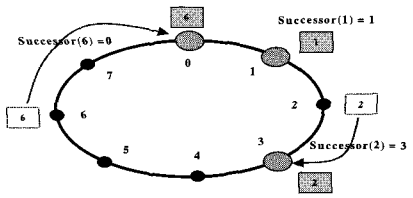
1. Chord

Chord[4]는 2001년 MIT와 U.C.Berkeley에서 처음 제안한 DHT 기반의 오버레이 네트워크 구축 방식이다. Chord는 노드와 key의 hash값을 위해 원형의 m-bit 식별자 공간을 갖는다. 즉, 각각의 노드와 각각의 key는 m-bit ID를 갖게 된다. 오버레이 네트워크 참여를 위한 Node ID는 IP 주소를 hashing한 m-bit의 값으로 구성하며, 각 파일의 경우 파일의 이름을 hashing한 m-bit 값이 key ID가 된다. 이 key의 정보는 successor node에 저장이 되는데, successor node란 원형의 식별자 공간에서, key와 같거나 더 큰 node ID를 갖는 첫 번째 노드를 말한다. [그림 3]에서 key ID 정보가 저장되는 successor node에 대한 사례를 보인다.

Chord 에서는 key 정보를 삽입하거나 key 정보를 소유한 검색 메시지를 해당 노드로 전달하기 위한 라우팅 테이블 역할을 하는 finger table을 유지하고 관리한다. finger table의 구성은 start, interval, successor로 구성된다. 표 1 에서 finger table의 구성을 위한 notation 을 볼 수 있다.

표 1에서 start 값은 노드의 위치를 기준으로 시계방향으로 거리를 exponential하게 증가시켜 가면서 생성된 값이다. 이 값들을 기초로 interval 값과 successor 값이 결정된다. 또한 오버레이 네트워크를 유지하기 위해 predecessor에 대한 정보도 유지한다. 그림 4에서는 1번 노드에 대한 start와 interval 값의 구성을 보여준다. start 값이 exponential 하게 증가되는 형태로 구성되므로 실제 검색 수행 시 평균 log(n) time lookup이 가능 하다. 그림 5에서 실제 finger table의 구성사례를 볼 수 있다.

그림 5에서는 3번 노드가 key 1에 대한 정보를 얻기 위한 검색과정을 볼 수 있다. 3번 노드에서 key 1은 자신의 finger table 중 interval [7,3)에 속하므로 해당 entry(table의 세 번째 entry)의 successor 노드인 0번 노드로 검색 메시지를 전달한다. 이 검색 메시지를 수신한 0번 노드는 같은 방식으로 결정된 successor 노드인 1번 노드로 이 검색 메시지를 전달한다. 1번 노드는 자신이 key 1에 대한 정



[그림 2] DHT기반 오버레이 네트워크

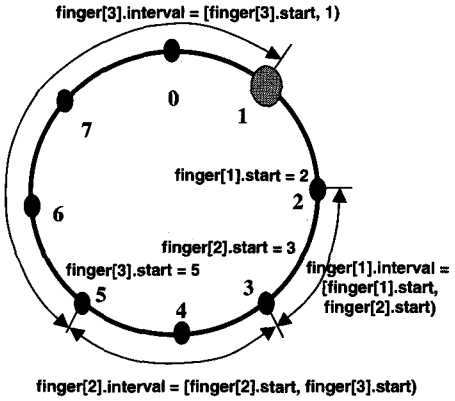
[그림 3] Successor Node

〔표 1〕 Finger table notation 정의 (m-bit 식별자 공간을 갖는 node n)

Notation	Definition
finger[k].start	$(n+2^{k-1}) \bmod 2^m$, $1 \le k \ge m$
.interval	*[finger[k].start, finger[k+1].start)*
.node	first node ≥ *n.finger[k].start*
successor	the next node on the identifier circle; *finger[1].node*
predecessor	the previous node on the identifier space

보를 가지고 있으므로 그 정보를 노드 3에게 반환한다. 실제 P2P 응용에서 이 때 반환되는 정보는 key와 value 즉 IP 주소가 된다.

그림 6은 새로운 node(6번 노드)가 오버레이 네트워크에 참여하기 위해 join 한 이후의 Finger table과 key 정보 보유에 대한 내용을 보여준다. 실제 Join하는 노드는 자신의 successor와 preseceesor를 비롯한자신의 finger table 구성을 위한 정보를 다른 노드로부터 수신하고, 자신의 정보도 전달하는 과정을 통해 자신과 주변 노드의 finger table을 갱신하고 key를 transfer 하는 과정이 진행된다. 또한 자신이 보유하고 있는 파일의 위치 정보를 알리기 위해 hash 함수를 돌려 구한 key ID값들을 알고리즘에 따라 결정된 다른 node에게 전달해 주는 과정도 필수적이다. node가 오버레이 네트워크에서 탈퇴하는 경우고 비슷한 역과정을 거친다. 그림 7은 1번 노드가 탈퇴한 경우에 대한 finger table의 key 보유 정보를 보여준다.



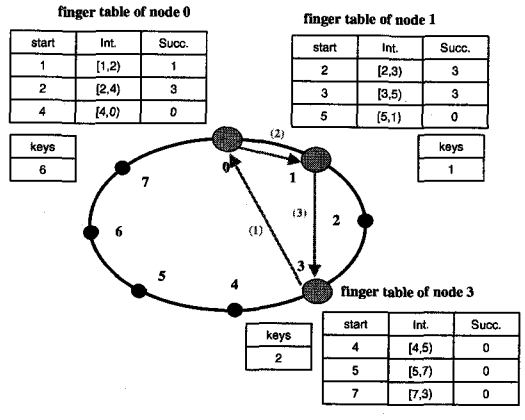
〔그림 4〕 Start & Interval

Chord의 경우 노드의 failure가 발생하는 경우를 대비하여, r immediate successor-list를 보유하는 방안, (key, IP address)쌍을 중복하여 분산시키는 방안, 주기적으로 node가 살아있는 상태인지 refresh 메시지를 전송하는 방안 들이 제안되고 있다.

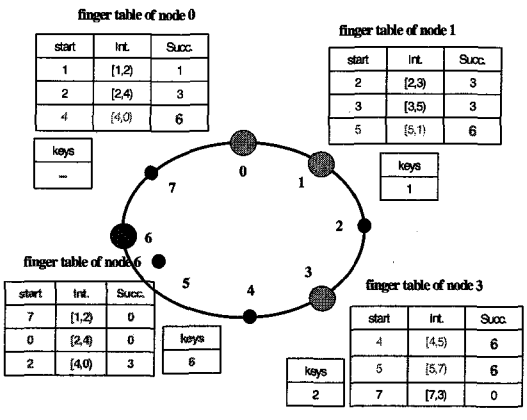
Pastry[5]와 Tapestry[6]의 경우에도 원형공간에서의 오버레이 네트워크 구축방식을 사용하는 면에서 Chord와 유사한 구조를 갖는다. 그러나 라우팅 방식, 테이블 구성 방식 등에 있어서 차이점이 존재한다. 예를 들어 Pastry의 경우엔 라우팅을 위해 Prefix matching 방식을 사용한다. Pastry[5]와 Tapestry[6]의 자세한 내용은 본 고에서는 생략한다.

2. CAN

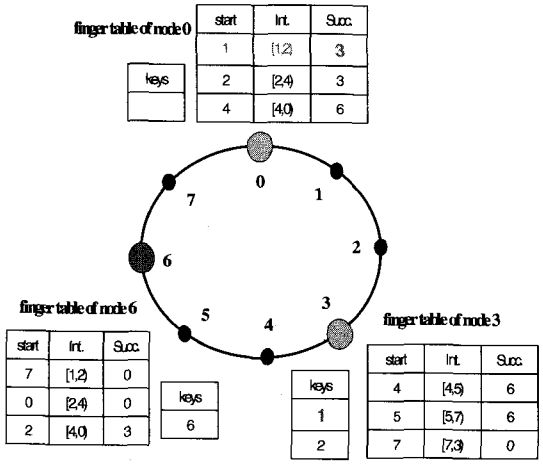
CAN[7,8](Content Addressable Network)은



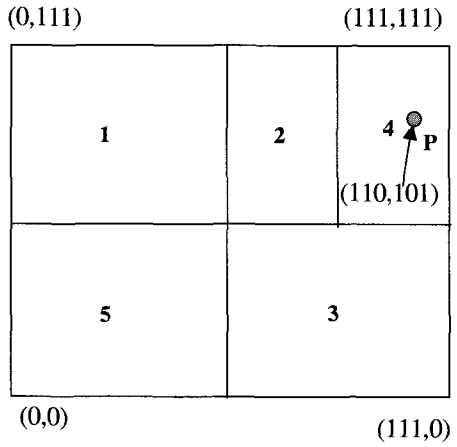
〔그림 5〕 Chord구조의 finger table과 routing



〔그림 6〕 노드 6번이 Join한 이후의 finger table



〔그림 7〕 노드 1번이 Leave한 이후의 finger table



〔그림 8〕 CAN의 2차원 공간

2001년 ACIRI(AT&T Cente for Internet Research)와 Berkeley 대학에서 제안한 DHT 기반 오버레이 네트워크 구축기법이다. Chord나 Pastry, Tapestry는 원형의 공간을 이용하여 오버레이 네트워크 구축을 하는 구조를 갖는 반면, CAN은 오버레이 네트워크 구축을 위해 m차원의 공간을 갖는다. Hash 함수를 통해 생성한 key 값도 chord의 경우에는 m-bit의 숫자였지만, CAN의 경우에는 m차원의 한 점에 대응하는 벡터 값을 갖는다. 그림 8은 2차원의 CAN 공간을 보여준다. 그림 8에서 (110,101)이라는 key의 정보(key 값, IP 주소)는 해당 zone을 소유하고 있는 4번 노드에 저장하게 된다. 그림 8의 경우 5개의 노드가 오버레이 네트워크에 참여하고 있으며, 각 노드는 자신의 neighbor set을 유지한다. 자신의 neighbor란 두 개 노드가 속한 좌표 값 중 d-1 차원의 값이 중복되면 서로 neighbor라고 한다. 그림 8의 경우 2번 노드의 neighbor는 1, 3, 4번 노드가 된다. 각 노드는 자신의 neighbor set을 알고 있으므로 라우팅은 좌표를 기반으로 자신의 neighbor 중 한 노드에게 전달하는 방식의 반복으로 동작하게 된다.

CAN에서 새로운 node가 join하는 과정은 다음과 같다.

(1) CAN 네트워크에 현재 참여하고 있는 노드를 검색한다. 이 과정은 응용에 따라 방법을 달리할 수 있다. CAN의 설계팀은 CAN 에 현재 참여하는 노드에 대한 정보를 갖고 있는 bootstrap node가 존재한다고 가정하여, 새로운 노드는 bootstrap node에게 문의하여 현재 참여 노드의 IP를 검색하는 방법을 사용한다.

(2) CAN 공간에서 임의의 좌표 P를 선택한 후, Join 메시지와 함께 (1)에서 발견한 노드에게 전송한다.

(3) CAN 노드들은 이 메시지를 해당 좌표에 도달할 때 까지 전달한다.

(4) 해당 zone의 소유자 node는 Join 메시지를 수신한 후, 자신의 zone을 두개로 분할 한 후, 새로운 노드에게 할당하여 준다.

(5) 이 때, 기존 노드는 새로운 zone에 해당하는 (key, value)쌍을 새로이 참여한 노드에서 전달해 준다.

(6) 기존 노드와 신규 노드 모두 neighbor set을 갱신한다.

(7) 기존 노드와 신규 노드의 neighbor들 모두 자신의 neighbor set을 갱신한다.

CAN에서 노드가 탈퇴하는 경우에는 자신의 neighbor 중 하나에게 자신의 zone과 자신이 보유하고 있는 (key, value) 쌍들을 전달하는 과정이 필요하며, 탈퇴한 노드의 neighbor들은 자신의 라우팅 테이블과 neighbor set을 갱신하게 된다. 노드의 failure가 발생하는 경우를 대비해서는 주기적인 beacon 메시지 생성 및 전송, take over node를 이용한 recovery 알고리즘, key 중복저장을 위한 multiple hash function 등의 방식이 제안되어 있다. 이에 대한 자세한 과정은 [7]을 참조할 수 있다.

현재 DHT 기반 오버레이 네트워크를 기반으로 하는 프로젝트가 일부 존재한다. 표 2에서는 현재 진행중인 관련 프로젝트를 정리하였다.

〔표 2〕 관련 프로젝트

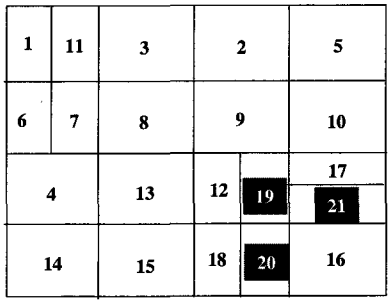
Project	기관	Overlay Network	분야
CFS	MIT	Chord	Distributed storage
PAST	MS, Rice Univ.	Pastry	Distributed storage
SCRIBE			Event Notification, Group Communication
Split-Stream			High-bandwidth Content distribution
POST			Co-operative messaging
Squirrel			Co-operative web caching
Scrivener			File Sharing
Herald	MS	Pastry	publish/subscribe event notification
DSPR	purdue	Pastry 확장	Mobile Ad-hoc routing
Ocean-store	U.C. Berkeley	Tapestry	distributed storage

Ⅲ. DHT 기반 오버레이 네트워크 보안 위협

본 절에서는 DHT 기반의 오버레이 네트워크를 구축하여 P2P 서비스를 제공하는 경우에 발생할 수 있는 보안 취약성[11,12]을 분석한다.

1. Node ID 관련 보안 위협

P2P 오버레이 네트워크에에 참여하고자 하는 피어는 자신의 Node ID 즉 Joining point를 선택할 수 있으며, 이로 인한 많은 보안 위협이 존재하게 된다. 예를 들어 그림 9와 같은 CAN 네트워크를 가정해 보자. 만약 공격자가 Joining point로 19, 20, 21를 선택하여 Join을 했다고 하면, 이 공격자는 16번 노드의 오버레이 네트워크 접근을 제어할 수 있게 된다. 또한 다른 노드에서 16번 노드가 관리하는 정보를 얻기 위해 접속을 시도할 경우, 이 역시 공격자에 의해 제어될 수 있게 된다. 또한 공격자는 파일에 대한 접속을 제어할 수도 있다. 그림 9에서 파일 A에 대한 정보는 A를 hashing 하여 결정된 좌표의 zone을 포함하는 노드가 관리하게 되는데 예를 들어 그 노드가 8번 노드라면, 공격자는 8번 노드의 위치를 선택하여 Join 메시지를 보내면 50%의 확률로 그 파일의 정보에 대한 소유자가 될 수 있으며, 이를 통해 파일의 접근을 제어할 수 있게 된다. 이와 관련된 공격으로 Sybil Attack이 있다. Sybil Attack은 공격자가 많은 수의 합법적인 노드 ID를 획득하여 공격에 사용하는 것이다. IPv6 네트워크라면 공격자는 더 많은 IP 주소 획득이 가능하므로, 이 문제는 더욱 심각해 질 수 있다.



〔그림 9〕 Node ID 관련 공격 사례

Chord와 Pastry 같은 경우에도 자신의 노드 ID를 선택할 수 있다면, ID circle 상에 joining point를 선택할 수 있기 때문에, CAN 에서와 동일하게 P2P 오버레이 네트워크 접속과 특정 파일에 대한 접속을 임의대로 제어할 수 있는 위협이 존재하게 된다. 특정 joining point를 선택할 수 없는 경우라도, 다수의 Node ID를 생성하여 공격하는 Sybil Attack을 통한 위협은 여전히 존재할 것이다.

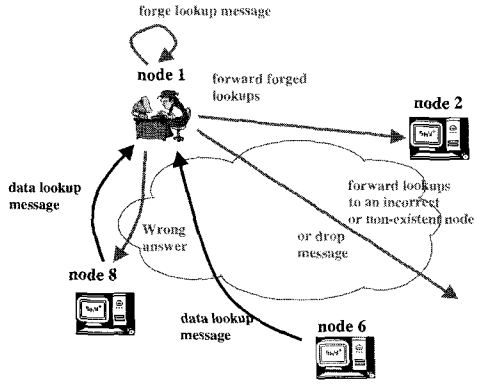
2. 라우팅 관련 보안 위협

앞선 장에서 설명된 바와 같이 DHT 기반 P2P 오버레이 네트워크에서는 각각의 피어들이 응용계층에서 검색 메시지를 전달하는 라우팅에 참여하게 된다. 라우팅 관련 보안 공격은 메시지를 라우팅하는 과정에 참여하는 노드의 다음과 같은 행동으로 야기된다.

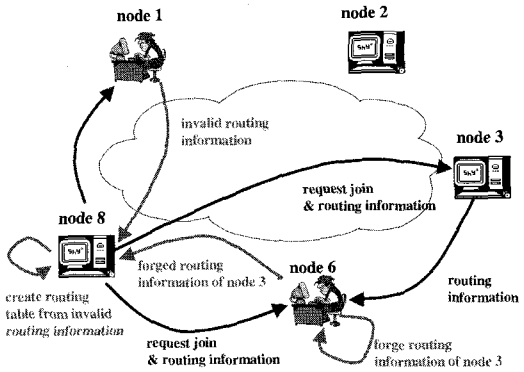
- 메시지를 전달하지 않고 임의로 폐기
- 메시지를 잘 못된 위치로 전달
- 메시지를 위변조하여 전달
- 거짓 정보를 반환

그림 10는 상기의 행동으로 인한 라우팅 보안 위협을 보여준다.

또한 라우팅 관련 공격으로 노드들의 라우팅 테이블을 잘못된 정보로 채우도록 유도하여 오버레이 네트워크 자체의 동작을 방해하는 공격이 있다. DHT 기반의 P2P 오버레이 네트워크에서는 새로운 노드가 Join하는 경우 오버레이 네트워크 구축 알고리즘에



〔그림 10〕 라우팅 방해 공격



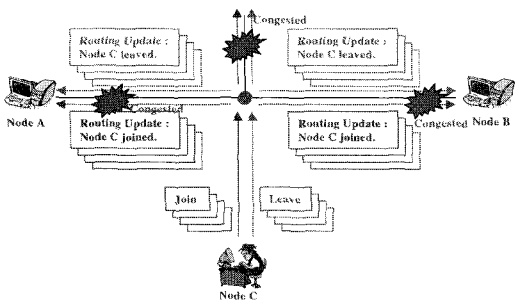
〔그림 11〕 라우팅 테이블 관련 공격

의해 결정된 피어 간에 라우팅 테이블의 내용 교환을 통해 자신의 라우팅 테이블을 갱신하는 구조를 갖는다. 따라서 자신의 라우팅 테이블 또는 전달해야 하는 라우팅 테이블 정보를 위변조 한다면, 오버레이 네트워크가 제대로 동작할 수 가 없게 될 것이다. 그림 11에서 이러한 라우팅 테이블 관련 위협을 보여준다.

3. DoS 공격

DHT 기반의 오버레이 네트워크에서 DoS(Denial of Service) 공격은 다양한 방법으로 가능하다. 공격자가 수많은 검색 메시지를 생성하여 희생자 노드에게 전송하는 방법이 가능하다. 또한 공격자들이 자신이 전달하는 검색 메시지를 모두 희생자 노드에게 전송하거나 검색 요구가 있을 때 모두 그 소유자가 희생자 노드인 것으로 반환을 한다면, 희생자 노드로 트래픽이 집중될 것이다.

또 하나의 심각한 위협 중 하나는 바로 Rapid Join and Leave 공격이다. DHT기반의 오버레이 네트워크에서는 새로운 노드가 참여하거나 나갈 때 이웃 노드들과 라우팅 테이블을 비롯한 각종 정보교환을 통해 각 노드의 라우팅 테이블을 갱신하게 되는데, 짧은 간격을 두고 Join과 Leave를 반복하게 되면 이를 처리하기 위해 네트워크가 폭주되어 DoS 공격의 효과를 노릴 수 있게 된다. 그림 12는 Rapid Join and Leave 공격의 사례를 보여준다.



〔그림 12〕 Rapid Join and Leave 공격

상기 언급한 보안 위협 외에도 P2P 오버레이 네트워크의 정상 참여자가 정의된 프로토콜을 준수하지 않고 비정상적인 동작을 통해 야기되는 다양한 보안 공격이 가능할 것이다. 또한, 오버레이 네트워크 참여자가 아닌 노드에 의한 도청, IP Spoofing 공격등도 가능한 위협이 될 수 있다.

본 고에서 언급한 보안 위협은 DHT기반으로 구축된 P2P 오버레이 네트워크에 특화된 보안 위협이며, 실제로 일반 P2P 서비스에 대한 보안 위협들 - 기밀정보 유출, 저작권 침해, Worm/Virus의 전파, System Hacking, Spoofing 등 - 역시 함께 존재한다.

Ⅳ. 결 론

본 고에서는 최근 연구가 많이 진행 중인 DHT 기반의 P2P 오버레이 네트워크의 구조와 이에 대한 보안취약성을 분석하였다.

본 고에서 언급한 보안 취약성에 대한 다양한 대응 기술도 제안되고 있다. Node ID 관련 공격에 대응하기 위한 방안으로 Node ID 생성 시 IP 주소와 공개키를 사용하는 방안, 노드 ID를 인증할 수 있는 Trust Authority(CA)서버를 두는 방안 등이 있다. 또한 Sybil attack을 방지하기 위해 노드 ID 당 요금을 받아 공격자의 진입속도를 늦추거나 다수의 ID

획득을 방지하는 방안, 노드 ID와 real world ID를 mapping 시키는 방안도 가능하다. 라우팅 관련 공격의 경우, 다중 hash 함수를 사용하여 key 정보를 복제하여 분산시키고 이에 대한 라우팅 경로를 다양화하는 방안 등도 제시되고 있다. Pastry 기반의 프로젝트인 PAST[14]의 경우엔 PKI 기반의 전자서명(Digital Signature)을 통한 인증과 파일정보 복제 및 분산 기법을 채택하고 있다. 또한, DoS 공격을 방지하기 위한 Processing Seheduling 방식, Rapid Join and Leave 공격을 방지하기 위한 P2P 네트워크 진압과 탈퇴를 위한 시간 제약을 두는 방안 등도 제안되고 있다.

현재까지 DHT 기반의 오버레이 네트워크를 인터넷 범위의 P2P 서비스에 실제 응용되고 있지는 않으며, 특정 도메인 내에서 분산 스토리지 시스템 등의 응용에 활용되고 있다. DHT 기반의 오버레이 네트워크는 검색 효율성 측면에서 매우 뛰어나기 때문에 보안문제가 해결 된다면 차세대 P2P 서비스로의 활용을 충분히 고려할 수 있을 것이다.

참 고 문 헌

[1] A. Parker, "The Impact of P2P", NGN (Next Generation Networks)2005 presentation slides, Sep. 26-30, Washingon D.C., 2005.

[2] Napster, http://www.napster.com

[3] The Gnutella Protocol Specification v0.4, Document Revision 1.2

[4] Ion Stoica, Robert Morris, David Karger, M.Frans Kaashoek, Hari Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," ACM SIGCOMM'01, 2001

[5] Antony Rowstron and Peter Druschel, "Pastry:Scalable, decentralized object location and routing for large-scale peer-to-peer systems," Proc. of the 18th IFIP/ACM International Conference on Distributed Systems Platforms(Middleware 2001). Heidelberg, Germany, Nov. 2001

[6] D.Barkai, "Peer-to-Peer Computing : Technologies for Sharing and Collaborating on the Net," Intel-Press, 2002

[7] S. P. Ratnasamy, "A Scalable Content Addressable Network" Ph.D thesis, University of Berkeley, fall 2002

[8] S. Ratnasamy, P.Francis, M.Handley, R.Karp, "A Scalable Content-Addressable Network", SIGCOMM'01, 2001

[9] Chord, http://pdos.csail.mit.edu/chord/

[10] PASTRY, http://freepastry.rice.edu/

[11] E. Sit and R. T. Morris, "Security Considerations for Peer-to-Peer Distributed Hash Tables", In the proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS '02), March, 2002; Cambridge, MA

[12] M.Castro, P.Druschel, A.Ganesh, A.Rowstron and D.S.Wallach,"Secure Routing for Structured Peer-to-Peer Overlay Networks", Proc. of the 5th Usenix Symposium on Operating Systems Design and Implementation, Boston, MA, Dec., 2002.

[13] CFS project, http://pdos.csail.mit.edu/chord/

[14] PAST project, http://freepastry.rice.edu/PAST/default.htm

[15] SCRIBE project, http://freepastry.rice.edu/SCRIBE/default.htm

[16] SplitStream project, http://freepastry.rice.edu/SplitStream/default.htm

[17] POST project, http://freepastry.rice.edu/POST/default.htm

[18] Squirrel project, http://freepastry.rice.edu/Squirrel/default.htm

[19] Scrivener project, http://freepastry.rice.edu/Scrivener/default.htm

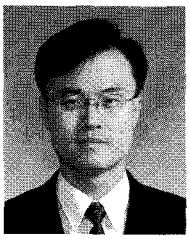
[20] Herald project, http://research.microsoft.com/sn/Herald/

[21] DSPR project, http://dynamo.ecn.purdue.edu/~ychu/projects/dpsr/

[22] Oceanstore project, http://oceanstore.cs.berkeley.edu/

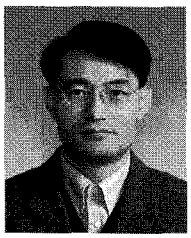
[23] Y.D.Kim, P2P networks and their security (presentation slides), ETRI 전문가 초청 세미나, 2005

〈著 者 紹 介〉



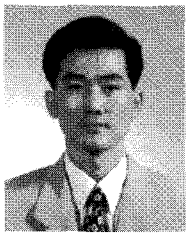
권 혁 찬 (Hyeok Chan Kwon)

1994년 2월 : 서원대학교 전자계산학과 졸업
1996년 2월 : 충남대학교 전산학과 석사
2001년 2월 : 충남대학교 컴퓨터과학과 박사
2001년 1월~현재 : 한국전자통신연구원 P2P보안연구팀 선임연구원
〈관심분야〉 네트워크 보안, IPv6 보안, P2P 보안



나 재 훈 (Jae Hoon Nah) 정회원

1985년 : 중앙대학교 컴퓨터공학과 졸업
1987년 : 중앙대학교 컴퓨터공학과 석사
2005년 : 한국외국어대학교 전자정보공학과 박사
1987년~현재 : 한국전자통신연구원 P2P보안연구팀 팀장
〈관심분야〉 IPv6/MIPv6 보안, P2P 보안



장 종 수 (Jong Soo Jang) 정회원

1984년 : 경북대학교 전자공학과 공학사
1986년 : 경북대학교 전자공학과 공학석사
2000년 : 충북대학교 컴퓨터공학과 공학박사
1989년~현재 : 한국전자통신연구원 정보보호연구단 네트워크보안그룹 그룹장
〈관심분야〉 네트워크보안, 웹서비스보안, Secure OS, IDS/IPS, Traffic Management