IP 망에서의 액티브 노드 발견 및 액티브 패킷의 신뢰성 전송 기법

김 방 은[†] · 채 기 준[††] · 김 동 영[†††] · 나 중 찬[††††]

요 약

IP 망에 액티브 네트워크 기술을 도입하기 위해서는 네트워크 내에서 물리적으로 직접 연결되지 않은 액티브 노드들이 서로에 대한 토폴로지 정보를 구성하고 관리하는 방법이 필요하다. 또한 하나의 프로그램을 수행하기 위해 필요한 액티브 패킷들은 손실 없이 전송되어야만 정상적인 작업을 수행할 수 있다. 뿐만 아니라 액티브 패킷은 중간 노드에서의 실행으로 기존의 네트워크에 비해 큰 융통성을 가지므로 효율성과 안전성이 보장되어야 한다. 본 논문에서는 네트워크 내의 액티브 노드들이 상호 간에 가입과 탈퇴 여부를 발견하기 위해서 OSPF(Open Shortest Path First) 프로토콜의 Opaque LSA(Link State Advertisement)를 이용하도록 하였다. 또한 수신지에서 손실을 감지하여 송신지에 재전송을 요구하는 신뢰성 전송 방식과 액티브 패킷의 사본을 이용한 복사 후 전송, 송·수신지에서의 보안 기법 적용이 이루어지도록 하는 액티브 패킷 전송 엔진을 제안하였다. 시뮬레이션을 통하여 기존의 기법과 비교·분석한 결과, 적용한 액티브 노드 발견 기법과 액티브 패킷 전송 엔진이 효율적임을 확인할 수 있었다.

Discovery of Active Nodes and Reliable Transmission of Active Packets in IP Networks

Bang-Eun Kim[†] · Ki-Joon Chae[††] · Dong-Young Kim[†††] · Jung-Chan Na[††††]

ABSTRACT

All active nodes which have no physically direct connection with each other in IP network must be able to compose and manage network topology informations. Besides one active program can be performed by the active nodes when every active packet for this program is transmitted without any loss of packets. Also the active packets should be transmitted effectively to minimize the transmission delay and securely from threatens. In this thesis, the discovery scheme of active nodes is adapted for active nodes in IP networks to compose and manage the topology information. The scheme for the efficient, reliable and secure transmission of active packets is also proposed. The sequence number is assigned to every active packet. If a receiver detects the loss of active packet checking the sequence number, the receiver requests the retransmission of the lost packet to the previous active node. After receiving an active packet and adapting security and reliability schemes, intermediate active nodes not only copy and send the packet instantly but also apply some security mechanisms to it. And the active packet transmission engine is proposed to provide these transmission schemes. The simulation of the adapted active node discovery scheme and the proposed active packet transmission engine is performed. The simulation results show that the adapted active node discovery scheme is efficient and the proposed active engine has the low latency and the high performance.

키워드 : 액티브 네트워크(Active Network), 액티브 노드 발견(Active Node Discovery), 액티브 패킷 전송(Active Packet Transmission)

1. 서 론

액티브 네트워크(Active Networks)란 라우터나 스위치가 프로그램 실행 능력을 가지고 있어서 프로그램을 포함하고 있거나 중간 노드의 프로그램을 실행하도록 하는 패킷을 처리하여 다양하고 유동적인 처리를 패킷에 행할 수 있는 환경을 가진 망을 말한다[1]. 이러한 액티브 네트워크의 기술은 종단에서만 수행하던 네트워크의 기능을 분산시키고 사용자의 망에 대한 요구 기능을 시기 적절하게 반영할 수 있다. 그러나 IP 망의 중간에 위치한 액티브 라우터는 이웃 액티브 라우터들과 물리적으로 직접 연결되어 있지 않기 때문에 액티브 패킷을 전달할 다음 액티브 노드를 선택하기 위한 방법이 필요하다. 현재 액티브 네트워크에 관련된 대부분의 연구에서는 액티브 네트워크 토폴로지를 정적으로 구성하거나 혹은 구성되어 있다고 가정한다. 정적

※ 본 연구는 2002년도 한국전자통신연구원 정보보호연구단의 위탁연구 과제에 의한 것임.
† 정 회 원 : 이화여자대학교 대학원 컴퓨터학과
†† 종신회원 : 이화여자대학교 컴퓨터학과 교수
††† 정 회 원 : 한국전자통신연구원 정보보호연구단 연구원
†††† 정 회 원 : 한국전자통신연구원 정보보호연구단 책임연구원
논문접수 : 2003년 4월 21일, 심사완료 : 2004년 3월 8일

인 네트워크 구성 방법은 구현은 용이하나 네트워크의 동적인 변화를 반영하지 못하고 관리자의 많은 노력을 필요로 한다. 따라서 액티브 노드가 동적으로 다른 액티브 노드들의 토폴로지 정보를 알 수 있는 메커니즘이 필요하다. 또한 특정 프로그램을 포함하고 중간의 액티브 노드에서 실행될 액티브 패킷들은 실행하고자 하는 액티브 노드로 손실 없이 전송되어야만 정상적인 작업을 수행할 수 있다. 그리고 중간 노드에서 프로그램이 실행되므로 패킷의 전송지연이 길어질 수 있다. 뿐만 아니라 액티브 패킷은 코드의 실행에 따라 라우터의 상태를 변경할 수도 있는 융통성을 지니므로 비인가자에 의한 도청, 패킷의 생성·변경·파괴 등의 위협에 대하여 안전성이 보장되어야 한다. 그러므로 기존의 IP 망에 도입이 용이한 액티브 노드 발견 방법과 빠르고 신뢰성이 있으며 안전한 액티브 패킷 전송 기법에 대한 연구가 필요하다.

본 논문에서는 기존의 라우터와 액티브 라우터가 혼합되어 있는 IP 망에서 동적으로 액티브 노드를 발견하는 기법을 도입한다. 또한 액티브 패킷의 전송에 효율성과 신뢰성, 안전성을 보장하는 기법을 제안하고 이를 이용하여 액티브 패킷 전송 서비스를 제공하는 액티브 패킷 전송 엔진을 제안한다. 그리고 적용한 액티브 노드 발견 기법과 액티브 패킷 전송 엔진을 모델링하여 성능을 측정하고 기존의 방법들과 비교·분석한다.

본 논문의 구성은 다음과 같다. 1장의 서론에 이어 2장에서는 액티브 노드의 발견과 액티브 패킷의 전송에 관련된 기존의 연구에 대하여 살펴본다. 3장에서는 액티브 노드 발견을 위해 적용한 기법을 설명한다. 또한 액티브 패킷 전송을 위해 제안한 기법과 액티브 패킷 전송 엔진을 설명한다. 4장에서는 액티브 노드 발견 기법과 액티브 패킷 전송 엔진을 시뮬레이션하여 중앙 집중화된 액티브 네트워크 토폴로지 구성 방식 및 기존의 전송 프로토콜과 성능을 비교하고 분석한다. 마지막으로 5장에서 본 논문의 결론을 제시한다.

2. 관련 연구

이 장에서는 액티브 네트워크 토폴로지 구성 방식과 액티브 패킷 전송의 신뢰성 기법에 관련된 연구를 살펴본다.

2.1 액티브 네트워크에서의 노드 발견에 관한 연구

액티브 네트워크에서 제안된 네트워크 토폴로지 구성 및 라우팅 프로토콜로 PLAN(Programming Language for Active Networks) 프로젝트에서 제안하는 방법에 대해 살펴본다. 그리고 IP 망에서 액티브 노드들에 대한 상태 정보를 얻기 위한 방안으로 OSPFv2의 Opaque LSA Option을 이용하는 방법에 대해 살펴보겠다.

2.1.1 PLAN 프로젝트 - SLRP(Service Layer Routing Protocol)

벨코어와 펜실베니아 대학에서 공동으로 연구하고 있는 SwitchWare 프로젝트에서 네트워크 API로 사용하기 위한 스크립트 언어인 PLAN[2]는 액티브 패킷의 유연성 있는 경로 지정을 위하여 SLRP[3]을 이용하였다. SLRP는 라우팅 정보를 제어하기 위하여 사용하는 동적인 경로 설정 프로토콜로 중앙 집중화된 방식을 이용한다. 전체 네트워크 내에서 하나의 노드를 마스터 노드(Master Node)로 정하고 나머지 다른 노드들은 마스터 노드의 종속 노드(Slave Node)로 설정한다. 종속 노드는 호스트와 호스트를 연결하는 라우팅 테이블을 가지고 있으며, 주기적으로 이웃 노드들에게 패킷을 보내어 일정 기간동안 응답이 오지 않으면 마스터 노드에게 이웃 노드가 네트워크로부터 탈퇴하였음을 알리는 메시지를 보낸다. 마스터 노드는 종속 노드로부터 보고 받은 정보를 바탕으로 라우팅 그래프를 다시 계산하고 각 종속 노드들의 라우팅 테이블을 갱신하여 변경 메시지를 보냄으로써 전체 네트워크를 관리한다.

2.1.2 The OSPFv2 Opaque LSA Option

ALCATEL의 D. Galand와 O. Marce가 제안한 방식으로 액티브 노드의 위치와 액티브 노드가 포함하고 있는 실행 환경(EE : the Execution Environment)에 관한 정보를 전체 네트워크에 전달하기 위해서 OSPFv2의 Opaque LSA Option[4]를 이용하도록 한다[5]. 실행 환경이란 중간의 액티브 노드에서 액티브 패킷을 읽어 들여 처리할 수 있도록 제공하는 환경을 말한다. Opaque LSA Option은 OSPFv2에서 미래에 OSPF를 확장할 경우를 대비하여 제공하는 LSA 형태로서 네트워크를 통해 공유하고 싶은 정보를 정의해서 사용할 수 있다. 액티브 노드 정보를 위해 사용하도록 제안한 Opaque LSA의 형식은 (그림 1)과 같다.

<table>
<tr><td colspan="2">LS age</td><td>Options</td><td>9, 10 or 11</td></tr>
<tr><td>Opaque Type</td><td colspan="3">Opaque ID</td></tr>
<tr><td colspan="4">Advertising Router</td></tr>
<tr><td colspan="4">LS Sequence Number</td></tr>
<tr><td colspan="2">LS checksum</td><td colspan="2">Length</td></tr>
<tr><td>EEID</td><td>EE Info length</td><td colspan="2">EE Info</td></tr>
<tr><td colspan="4">EE Info(cont.)</td></tr>
</table>

(그림 1) Opaque LSA 형식

- Opaque Type : 액티브 라우터임을 나타낸다.
- Opaque ID : 이 LSA가 나타내고자 하는 액티브 라우터의 특징을 가리킨다. 현재 0이 실행 환경에 관한 정보를 포함하고 있다는 것을 나타낸다.

- EE ID : 실행 환경의 식별자로 ANANA(Active Networks Assigned Number Authority)[6]에서 지정하는 값이다.
- EE Info Length : 실행 환경에 관련된 부가 정보의 길이를 나타낸다.
- EE Info : 실행 환경에 관한 정보를 담고 있다.

나머지 필드들은 기본 LSA[7]의 형식과 같다.

2.2 액티브 네트워크에서의 신뢰성 전송에 관한 연구

현재 액티브 네트워크에서는 멀티캐스트 분야에서 신뢰성 전송과 관련한 연구가 진행되고 있다. 대표적인 멀티캐스트 환경에서의 신뢰성 전송에 관한 연구를 살펴본다.

2.2.1 ARM(Active Reliable Multicast)[8]

ARM은 MIT에서 신뢰성 있는 멀티캐스트 서비스를 제공하기 위하여 제안한 것이다. 송신자와 수신자의 종단간에서 일어나던 손실 복구 기법을 중간의 액티브 라우터와 수신자 사이에 적용시킴으로써 대역폭의 낭비를 막고 복구 지연을 줄일 뿐만 아니라 재전송의 부하를 분산시킨다. 또한 수신자가 수신 패킷의 일련번호를 검사하여 손실이 감지되었을 때에만 '부정 확인'을 이용하여 손실된 패킷의 재전송을 요구하도록 하므로 패킷을 수신할 때마다 전송하는 '긍정 확인' 기반의 방식보다 대역폭의 낭비가 훨씬 적다.

2.2.2 AER/NCA(Active Error Recovery/Nominee-based Congestion Avoidance)[9]

AER/NCA는 Tascnet의 PANAMA(The Protocols for Active Networking with Adaptive Multicast Applications) 프로젝트의 일부분으로 멀티캐스트 환경에서 효과적으로 손실 복구와 혼잡 제어를 하기 위해 제안된 방법이다. 손실 복구 방법인 AER은 ARM과 마찬가지로 '부정 확인' 기반의 지역 손실 복구 기법을 채택한다. 하지만 중간 라우터에서 손실 복구가 일어나는 ARM과 달리 액티브 복구 서버를 두고 중간의 라우터와 연결하는 방법을 이용한다. 이 방법은 ARM에 비하여 라우터의 변화를 요구하지 않기 때문에 기존의 IP 망으로의 적용이 쉽고 빠르다.

3. 액티브 노드 발견 기법 및 액티브 패킷 전송 기법

3.1 적용하는 액티브 노드 발견 기법

액티브 노드가 네트워크 내의 다른 액티브 노드들과 서로의 위치 및 실행 환경에 대한 정보를 교환할 수 있도록 하기 위하여 2장에서 살펴본 OSPF Opaque LSA Option을 이용하는 방법을 도입하였다. 이 때 사용하는 Opaque LSA의 형식은 2장에서와 거의 유사하다. 하지만 Opaque Type은 IANA[10]에서 실험적인 사용을 위해 제공하고 있는 범위의 번호인 128을 사용하도록 한다. Opaque ID에는 0을 넣어 Opaque 정보가 EE ID임을 나타낸다. 그리고 Opaque 정보 필드에는 액티브 노드 자신의 EE ID를 넣도록 한다. 나머지 필드들은 기본 LSA[7]과 동일하게 사용한다.

이 기법은 완벽하게 동적으로 작동하므로 네트워크의 동적인 변화를 인식시키지 못하고 관리자의 노력을 많이 요구하는 정적인 네트워크 토폴로지 구성의 단점을 극복할 수 있다. 뿐만 아니라 기존의 중앙 집중화된 방식을 이용하는 경우에 하나의 노드의 실패로 인해 네트워크가 마비될 수도 있는 위험에 대한 부담도 가지지 않는다.

3.2 제안하는 액티브 패킷의 전송 기법

액티브 패킷은 중간의 액티브 노드에서 수행되어야 하므로 손실 없이 전송되어야만 한다. 또한 액티브 패킷은 중간에서의 실행으로 인해 보안상의 위협이 크기 때문에 안전하게 전송되어야만 한다. 그리고 실행을 하더라도 패킷의 내용에 변화가 생기지 않는 경우에는 액티브 패킷이 액티브 노드에서 실행되는 시간뿐만 아니라 액티브 패킷들이 중간 노드에 모두 도착해서 다음 노드로 전송되는데 걸리는 전송 지연을 줄일 수 있어야 한다. 따라서 액티브 패킷을 효율적으로 안전하게 전송하면서도 부하가 작고 신뢰성 있게 전송하는 기법을 제안한다. 앞으로 '송신지'라는 표현은 액티브 패킷의 최초의 송신자 또는 받은 패킷을 전송하는 중간 액티브 노드를 말한다. '수신지'는 액티브 패킷의 최종 수신자 또는 패킷 전송 경로 상의 이전 노드로부터 패킷을 수신하는 중간 액티브 노드이다. 한편 하나의 프로그램을 수행하는데 필요한 액티브 패킷들이 같은 액티브 라우터를 경유하도록 하기 위해서 송신 경로 지정(Source Route)을 이용하는 것을 전제로 한다.

3.2.1 신뢰성 전송

대부분의 액티브 네트워크 연구에서는 전송 프로토콜로 UDP를 사용하고 있다. UDP는 전송을 위한 최소한의 기능만을 포함하므로 빠르고 가볍기 때문이다. 하지만 UDP는 신뢰성 있는 전송을 보장하지 못한다. 반면에 TCP는 신뢰성 있는 전송을 보장하지만 에러 제어, 흐름 제어, 폭주 제어 등의 서비스를 제공하기 때문에 부하가 크다. 따라서 UDP를 이용하면서도 신뢰성 있는 전송 기법을 제안한다.

(1) 손실 감지 기법

수신지가 들어오는 액티브 패킷의 일련번호를 검사하여 손실을 감지한다. 즉 패킷 손실 감지의 주체는 수신지가 된다. 수신지는 손실로 간주한 패킷에 대해 '재전송 요구 패킷'을 보내어 손실을 송신지에 알린다. '재전송 요구 패킷'은 손실된 패킷의 일련번호를 포함한다. 수신 측에서 들어온 액티브 패킷을 순서대로 정렬하고 손실을 감지할 수 있도록 하기 위해서 최초의 송신자는 액티브 응용으로부터 받은 액티브 메시지를 일정 크기의 블록으로 나누고 각각

에 대하여 일련번호를 할당한다. 이처럼 메시지를 일정 크기의 블록으로 나누는 것은 IP 계층에서의 조각화를 방지하기 위해서이다. 수신지가 들어오는 액티브 패킷들의 신뢰성 검사를 하기 위해서는 전송 받는 액티브 메시지에 대한 정보가 필요하다. 일련번호의 시작 번호나 전송될 패킷의 수 등에 대한 정보가 그것이다. 이러한 정보는 최초의 송신자가 보내는 메시지의 첫 번째 패킷에 실어 전송한다.

(2) 손실 복구 기법

송신지는 수신지로부터 '재전송 요구 패킷'을 받으면 패킷에 포함된 일련번호에 해당하는 액티브 패킷을 다시 전송한다. 수신지는 액티브 패킷을 받을 때마다 일련번호를 이용하여 신뢰성을 검사하고 손실을 감지하면 액티브 패킷을 전송한 바로 전 단계의 액티브 노드에 '재전송 요구 패킷'을 보내 재전송을 요청한다. 기존의 IP 망에서의 손실 복구는 종단간에 이루어지는 반면에 제안하는 기법에서는 액티브 패킷의 손실 복구가 홉 바이 홉으로 이루어지게 된다. 홉 바이 홉 손실 복구는 손실이 이루어진 영역 근처에서 복구가 이루어질 수 있으므로 재전송을 요구하는 패킷이나 재전송 되는 패킷들이 네트워크 전체를 돌아다니지 않아도 된다. 때문에 손실 복구로 인한 대역폭의 낭비와 같은 네트워크의 부하를 분산시킬 수 있다. 뿐만 아니라 손실 복구 시간도 줄일 수 있고 손실이 자주 일어나는 영역을 감지하는 데에도 이용할 수 있다.

제안하는 기법은 패킷 손실이 감지되었을 때에만 '재전송 요구 패킷'을 이용한다는 점에서 '부정 확인' 기반의 ARM[8], AER[9]의 방법과 동일하다. 하지만 기본적으로 멀티캐스트 트리의 중간 노드를 액티브 노드로 두고 데이터를 저장하도록 하여 유니캐스트에는 적용할 수 없었던 기존 연구의 제한점을 극복한 것이다.

3.2.2 복사 후 전송(Copy-and-Forward)

액티브 노드는 처리해야 할 액티브 패킷이 들어오면 패킷의 내용을 실행한 후에 다음 노드로 전송(저장-계산-전송)하므로 중간 액티브 라우터에서 지연이 발생하게 된다. 그리고 이 지연은 중간 액티브 노드의 개수가 늘어나고 전송하는 액티브 패킷의 양이 증가할수록 더욱 커지게 된다. 더욱이 지연은 액티브 패킷이 노드에서 실행되는 시간뿐만이 아니라 액티브 패킷이 노드에 모두 도착해서 다음 노드로 전송하는데 필요한 전송 지연도 포함한다. 하지만 실행을 하더라도 패킷의 내용에 변화가 생기지 않는 패킷의 경우에는 이러한 전송지연은 불필요하다. 따라서 본 논문에서는 중간의 액티브 노드에서 발생하는 지연을 줄이기 위하여 '복사 후 전송'하는 방식을 이용한다. '복사 후 전송'은 액티브 패킷이 모두 들어올 때까지 기다리지 않고 액티브 패킷이 들어오는 대로 사본을 남기고 원본은 다음 노드로 전송시키는 방식이다.

3.2.3 보안 서비스

송신자와 수신자 그리고 액티브 프로그램을 실행할 중간 노드들이 안전하게 통신하기 위한 방법으로 액티브 패킷에 보안 메커니즘을 적용하여 전송하도록 한다. 어떤 메커니즘을 적용할 것인가는 응용에서 선택하도록 한다. 제공하는 보안 서비스와 이를 위해 보안 메커니즘을 적용하는 방법은 아래와 같다.

(1) 기밀성(Confidentiality)

기밀성은 전송되는 정보의 불법적인 노출을 방지하는 기술이다. 기밀성은 액티브 패킷를 암호화하여 전송함으로써 제공할 수 있다. 송신지가 자신의 비밀키로 암호화하여 전송하면 수신 측에서 공유하고 있는 비밀키로 복호화한다. 또는 송신지가 수신지의 공개키로 메시지를 암호화하여 전송하면 수신지가 자신의 비밀키로 메시지를 복호화한다.

(2) 무결성(Integrity)

무결성은 전송되는 정보의 위조와 변조를 판단하는 기술이다. 무결성을 제공하기 위해서는 해쉬 함수를 이용한다. 송신지가 해쉬 함수를 이용하여 패킷 내의 데이터에 대한 메시지 축약(Message Digest)을 생성하고 이를 패킷에 붙여 전송한다. 수신지는 동일한 해쉬 함수를 액티브 데이터에 적용하여 메시지 축약을 생성하고 수신한 메시지 축약과 비교함으로써 송신 패킷 내 액티브 데이터의 위·변조 여부를 확인한다.

(3) 인증(Authentication)

인증은 정당한 송신지로부터 패킷이 전송되었는지를 입증하는 것이다. 인증은 공개키 기반의 암호화를 이용하여 제공하도록 한다. 송신지가 자신의 비밀키로 암호화한 서명을 메시지와 함께 전송하면, 수신지는 송신지의 공개키로 서명을 복호화하여 송신지의 정당성을 입증한다.

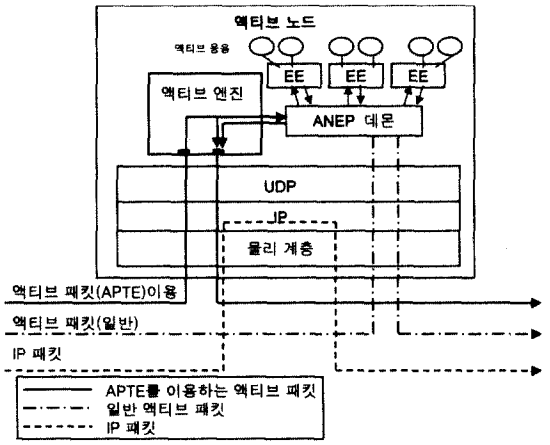
3.2.4 액티브 패킷 전송 엔진(APTE : Active Packet Transmission Engine)

네트워크에서 특정 작업을 수행하기 위해 필요한 일련의 액티브 패킷들에 신뢰성과 효율성, 안전성을 보장하는 기능을 제공하는 액티브 패킷 전송 엔진(APTE)을 제안하였다.

(1) 액티브 노드의 구조

액티브 노드에 설치될 제안한 APTE는 UDP와 ANEP (Active Network Encapsulation Protocol)[11] 사이에서 기능을 수행한다. UDP와 ANEP와는 포트를 통하여 액티브 패킷을 주고받는다. 이처럼 APTE를 사용자 레벨의 응용처럼 구현하고 포트를 할당하여 연결하는 이유는 구현이 용이할 뿐만 아니라 액티브 노드로 이식하기가 쉽기 때문이다[12]. 만약 UDP와 같은 전송 계층에서 액티브 패킷 전송 서비스를 제공하도록 구현한다면 속도는 좀 더 빨라질

수 있으나 커널 레벨에서 구현해야 하는 어려움이 따른다. 또한 액티브 노드로의 이식도 어렵고 보안상의 위협도 존재하게 된다. (그림 2)는 액티브 노드에서의 APTE의 위치와 패킷의 흐름을 나타낸 것이다.



(그림 2) 액티브 노드 구조와 액티브 패킷 전송 엔진(APTE)

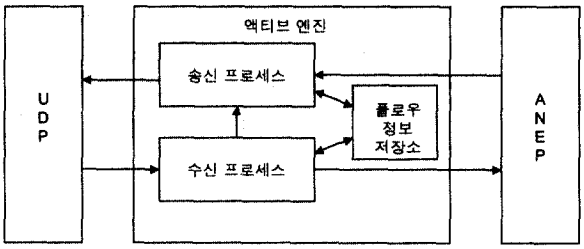
APTE가 제공하는 기능인 액티브 패킷 전송의 효율성 및 신뢰성, 안전성을 요구하는 패킷이 들어오면 UDP는 APTE로 액티브 패킷을 보낸다. APTE는 액티브 패킷의 안전성과 신뢰성을 검사한 다음에 복사한다. 만약 손실되었을 경우에는 '재전송 요구 패킷'을 보내고 안전성이 보장되지 않았을 경우에는 패킷을 폐기한다. 사본은 실행을 위해 ANEP 데몬으로 보내고, 원본은 다음 노드로 UDP를 통하여 전송한다. 액티브 패킷을 받은 ANEP 데몬은 액티브 패킷의 헤더를 보고 해당 실행 환경으로 전달한다. 실행 환경은 포트로 연결되어 있다.

만약 APTE 기능을 요구하지 않는 액티브 패킷이 들어오면 UDP는 그 패킷을 바로 ANEP 데몬으로 넘겨준다. IP 패킷이 들어오면 액티브 노드는 기존의 라우터에서와 같이 단순히 다음 노드를 결정하여 전달한다.

이와 같은 액티브 노드 구조에서는 다른 패킷들에게 영향을 끼치지 않으면서도 효율적으로 안전하게 손실 없이 전송되어야만 하는 액티브 패킷들에 대해 필요한 작업을 수행할 수 있다. 각 응용 프로그램들이 신뢰성과 안전성을 보장하는 기능을 중복적으로 구현하는 것은 시스템 자원의 낭비를 초래하기 때문에 APTE는 유용하다. 또한 기본 전송 계층인 UDP나 IP에 수정이 필요하지 않으므로 기존의 망에 적용하기가 용이하다. 뿐만 아니라 실행 환경보다 하위 계층에서 액티브 패킷에 보안 기법을 적용함으로써 보안의 강도는 높이고 노드의 자원은 절약할 수 있다.

(2) 액티브 패킷 전송 엔진(APTE)의 구조

APTE는 이전 노드로부터 액티브 패킷을 수신하였을 때에 동작하는 수신 프로세스와 액티브 패킷을 다음 노드로 전송하기 위한 동작을 수행하는 송신 프로세스, 들어오는 플로우들에 대한 정보를 저장하는 플로우 정보 저장소로 구성된다. 중간에서 특정 작업을 수행하는 액티브 노드들은 송신 프로세스와 수신 프로세스의 두 가지 기능을 모두 수행하여야 한다. 반면에 송신자는 송신 프로세스만을 수행하고 최종 수신자는 수신 프로세스만을 수행하면 된다. (그림 3)은 APTE의 구조를 나타낸다.



(그림 3) 액티브 패킷 전송 엔진(APTE)의 구조

각 프로세스가 수행하는 역할들은 다음과 같다.

- 송신 프로세스
  - 송신지로부터 액티브 패킷을 수신한 후에 송신자에서 액티브 노드 발견 기법에 의해 발견된 액티브 노드 정보를 바탕으로 설정한 전송 경로 상의 다음 액티브 노드 주소로 전송한다. 이때 안전성을 보장하기 위하여 전송 받은 데이터를 암호화하거나 해쉬 함수를 통해 메시지 축약을 생성하거나 또는 인증을 위한 암호화된 서명을 생성한다. 모든 액티브 패킷들은 플로우를 식별할 수 있도록 최초의 송신자 주소, 지정된 송신 경로와 최종 수신자 주소, 플로우 식별자 등을 항상 포함하고 있어야 한다. 특히 액티브 패킷의 최초 송신자는 액티브 메시지를 일정 크기로 자른 다음에 보안 메커니즘을 적용하고 일련번호를 할당하여 다음 액티브 노드로 전송한다. 이 때 첫 번째 전송 패킷에는 앞으로 송신하게 될 액티브 패킷들에 대한 정보, 즉 시작 일련번호와 패킷의 개수를 추가한다.
  - 만약 액티브 패킷을 전송 받는 다음 노드로부터 재전송 요구가 있을 경우에는 그 일련번호에 해당하는 액티브 패킷을 재전송 한다.
- 수신 프로세스
  - 액티브 패킷을 수신하면 '플로우 정보 저장소'로부터 정보를 얻어 일련번호를 검사한다. 수신한 액티브 패킷이 플로우의 첫 패킷일 경우에는 수신한 패킷으로부터 앞으로 수신될 액티브 패킷들에 대한 정보를 얻어 '플로우 정보 저장소'에 저장한다.
  - 손실이 감지되면 이전 액티브 노드로 '재전송 요구 패킷'을 전송한다. 손실이 없으면 액티브 패킷을 실행시키기 위하여 액티브 패킷의 사본을 ANEP로 전달한다.

- 액티브 노드가 중간 노드라면 다음 액티브 노드로 액티브 패킷을 전달하기 위하여 액티브 패킷을 송신 블록으로 보낸다.
- 수신한 액티브 패킷들의 순서를 정렬하고 중복된 액티브 패킷은 삭제한다.

● 플로우 정보 저장소

중간 액티브 노드로 들어오는 플로우들의 정보를 저장해 놓는다. 플로우는 액티브 패킷의 최초 송신자와 지정된 송신 경로 및 최종 수신자의 주소, 플로우 식별자 등으로 구분된다. 플로우 정보 저장소가 플로우별로 가지고 있어야 할 정보에는 수신할 액티브 패킷의 바로 다음 액티브 노드의 주소, 시작 일련번호, 전송될 패킷의 개수 등이 있다. 이 정보들은 액티브 패킷의 신뢰성과 안전성 검사에서 사용된다.

4. 성능 평가

이 장에서는 적용한 액티브 노드 발견 기법 및 제안한 액티브 패킷 전송 엔진(APTE)을 모델링하고 성능 평가를 수행한다. 데이터 전송량을 증가시키면서 전송지연과 처리율을 살펴본다. 중간 액티브 노드들의 개수에 따른 성능을 평가하고 분석한다. 또한 네트워크에 부하가 증가하여 에러가 발생했을 때의 전송지연과 처리율을 살펴보도록 한다. 시뮬레이션은 UCB(University of California, Berkeley)의 LBNL(Lawrence Berkely National Laboratory)에서 개발한 ns2[13]를 사용하고 토폴로지 생성을 위하여 Georgia Technologies에서 개발한 GT-ITM을 사용하였다. 또한 보안 메커니즘을 적용하기 위해서 다양한 암호 알고리즘을 제공하는 리눅스 버전의 Crypto++ 5.0[14]를 이용하였다.

4.1 시뮬레이션 환경

시뮬레이션의 환경은 GT-ITM으로 생성한 트랜짓-스텁(Transit-Stub) 구조로 현재의 인터넷 환경과 가장 유사하다[15]. 노드는 13개의 백본 노드를 포함하여 총 140개로 구성되어 있다. 링크는 99개의 LAN-LAN 링크, 41개의 MAN-WAN 링크, 22개의 WAN-WAN 링크들로 이루어져 있다. LAN-LAN 링크의 대역폭은 10Mbps, 지연은 2ms, MAN-WAN 링크의 대역폭은 8448Mbps(E2 캐리어), 지연은 20ms, WAN-WAN 링크의 대역폭은 34368Mbps(E3 캐리어), 지연은 30ms로 설정하였다[16].

시뮬레이션은 300초 동안 실행하였으며, 실제 인터넷과 비슷한 환경을 만들기 위하여 백그라운드 트래픽을 생성하였다. 백그라운드 트래픽으로는 전송 프로토콜로 TCP를 이용하는 HTTP, FTP, Telnet 트래픽과 UDP를 이용하는 CBR 트래픽을 사용하였다. 각 트래픽의 송신자와 수신자, 전송 시간은 임의로 결정하도록 하였다. 인터넷은 정확한 트래픽의 비율을 결정하는 것이 쉽지 않다. 본 논문에서는 "Wide-Area Internet Traffic Patterns and Characteristics"[17]을 참고로 하여 전체 플로우의 75%가 TCP, 25%가 UDP 패킷이 되도록 트래픽의 비율을 설정하였다. 그리고 TCP 패킷의 대부분은 웹 트래픽이고 FTP 플로우는 전체 플로우의 1% 이내가 되도록 하였다. <표 2>는 시뮬레이션에서 이용한 트래픽에 대하여 정리한 것이다.

액티브 노드 발견 기법의 성능 평가는 시뮬레이션 환경 중에서 13개의 노드, 10개의 LAN-LAN 링크와 4개의 MAN-WAN 링크로 구성된 하나의 자율 시스템을 대상으로 하였다. 그리고 적용한 액티브 노드 발견 기법과 성능을 비교하기 위하여 중앙 집중화된 모델을 구현하였다. 이는 2장에서 살펴본 SLRP와 유사하게 동작한다. 액티브 노드가 가입 메시지를 통해서 자신의 위치와 실행 환경 정보를 마스터 노드에게 알리면 마스터 노드가 자율 시스템에 있는 액티브 노드들에게 네트워크 내의 모든 액티브 노드에 대한 정보를 알린다. 하지만 액티브 노드가 이웃 액티브 노드의 변화를 감시하는 기능은 제외하고 탈퇴 시에는 탈퇴하는 노드가 탈퇴 메시지를 마스터 노드에게 보내면 마스터 노드가 자율 시스템 내의 모든 액티브 노드에게 알리는 것으로 단순화하였다.

APTE의 성능을 평가하기 위해서는 백그라운드 트래픽이 없을 때와 백그라운드 트래픽이 있을 때로 나누어서 시뮬레이션을 실행하였다. 백그라운드 트래픽이 없을 때의 테스트는 APTE가 에러가 없는 상황에서 정상적으로 작동하는 지와 APTE의 성능을 테스트할 수 있다. 백그라운드 트래픽이 있을 때의 테스트는 실제 인터넷과 비슷한 상황에서 혼잡 등의 이유로 에러가 발생하였을 때 APTE의 에러

〈표 2〉 시뮬레이션의 트래픽 설정

	백그라운드 트래픽				테스트 모델 (액티브엔진, TCP, UDP)
	HTTP	FTP	Telnet	UDP	
플로우 개수의 비율	28 (2800개부터 2, 3, 4배로 증가)	1 (100개부터 2, 3, 4배로 증가)	1 (100개부터 2, 3, 4배로 증가)	10 (1000개부터 2, 3, 4배로 증가)	20개로 고정
패킷 크기	임의로 결정	512bytes	73bytes	512bytes	512bytes
플로우당 전송량	0~300초 사이에 임의로 전송	0~200초 사이에 임의로 시작하여 1Mbytes 전송	0~200초 사이에 임의로 시작하여 100초 동안 전송	0~200초 사이에 임의로 시작하여 10초 동안 전송	0~200초 사이에 임의로 시작하여 1Mbytes 전송

복구 성능을 테스트할 수 있다. 제안한 APTE와 성능을 비교하기 위한 모델로는 TCP와 UDP를 채택하였다. 공정한 비교를 하기 위하여 동일한 양의 메시지를 같은 패킷 크기로 전송하였으며, 임의로 20개씩의 수신자와 송신자를 택하여 전송하도록 하였다. TCP는 최근 많이 이용하고 있는 Newreno 구현을 사용하였다.

4.2 시뮬레이션 결과 및 분석

이 절에서는 액티브 노드 발견 기법과 APTE의 성능을 테스트한 결과를 그래프로 나타내고 분석한다.

4.2.1 액티브 노드 발견 기법

적용한 액티브 노드 발견 기법의 성능을 자율 시스템 내의 액티브 노드의 개수를 늘려가면서 하나의 액티브 노드가 가입하고 탈퇴할 때 걸리는 시간을 측정함으로써 테스트하였다. 결과는 (그림 4)와 같다. 중앙 집중화된 방식(Centralized)보다는 본 논문에서 적용한 OSPF Opaque LSA를 이용하는 방식이 액티브 노드의 가입과 탈퇴를 더 빨리 발견할 수 있다. 중앙 집중화된 방식은 가입 혹은 탈퇴하는 노드가 우선 마스터 노드에 변경 메시지를 전송해야만 전체에 알릴 수 있기 때문에 마스터로의 메시지 전송 시간이 소요되어 가입, 탈퇴 노드 자신이 LSA 메시지를 직접 플러딩하는 OSPF Opaque LSA를 이용하는 방식에 비하여 발견 시간이 더 많이 걸린다. 또한 단문의 메시지 전송이므로 두 방법 모두 노드 수의 증가에는 비교적 영향을 받지 않는 것을 알 수 있다.

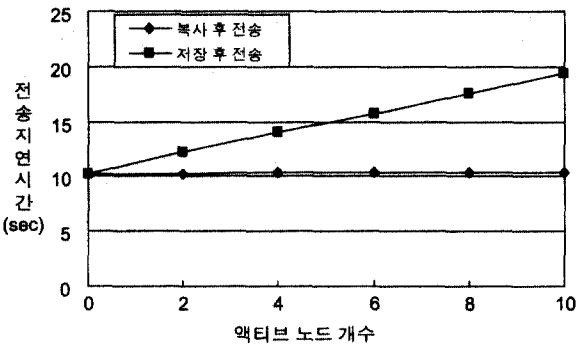
4.2.2 액티브 패킷 전송 엔진(APTE)

APTE의 성능을 백그라운드 트래픽이 없을 때와 있을 때로 나누어서 테스트한다. 백그라운드 트래픽이 없을 때에는 우선 '복사 후 전송' 방식의 성능을 테스트하고 다양한 보안 기법을 적용했을 때의 성능을 테스트한다.

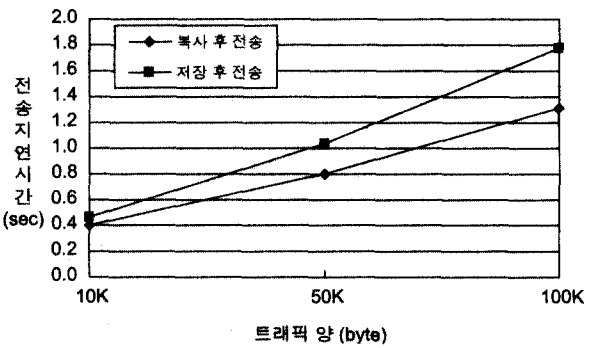
(1) 백그라운드 트래픽이 없을 때

● '복사 후 전송' 방식

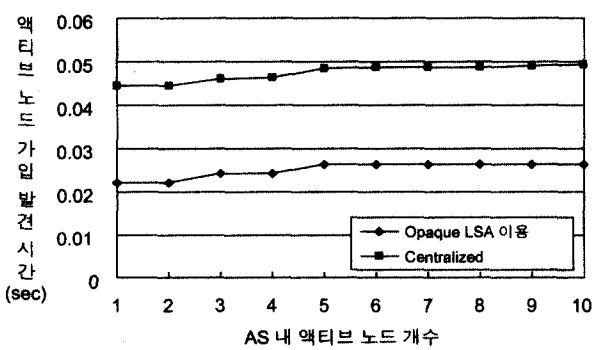
(그림 5)는 '복사 후 전송' 방식을 통하여 1Mbytes의 데이터를 전송할 때의 지연시간을 액티브 패킷이 거치는 중간 액티브 노드의 수를 증가시키면서 측정한 결과를 나타낸 것이다. (그림 6)은 트래픽 양을 변화시키면서 5개의 중간 액티브 노드를 거칠 때 '복사 후 전송' 방식의 전송지연시간을 측정한 결과이다. 이때 중간 액티브 노드는 임의로 결정하되 노드간에는 최소의 홉수가 되도록 경로를 지정하였다. 액티브 패킷을 모두 받을 때까지 기다려서 실행한 후에 다음 노드로 전송하는 액티브 네트워크에서의 기존의 방식과 비교하였다. 기존 방법에서의 실행 시간은 본 논문의 범위를 넘어서는 것이므로 기존 방법을 적용한 경우는 '저장 후 전송'이라 표기하였다. 보안 메커니즘은 적용하지 않았다.



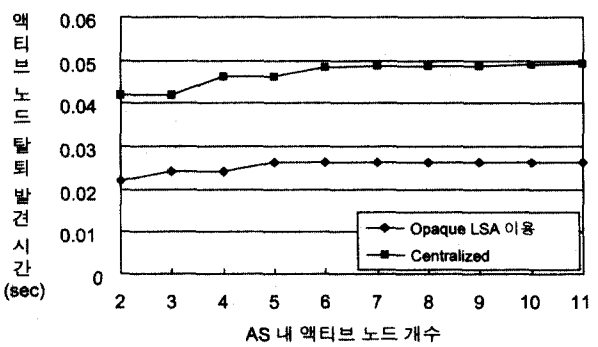
(그림 5) 노드 수에 따른 '복사 후 전송' 방식의 전송지연시간



(그림 6) 트래픽 양에 따른 '복사 후 전송' 방식의 전송지연시간



(a) 액티브 노드 가입 시



(b) 액티브 노드 탈퇴 시

(그림 4) 액티브 노드의 가입·탈퇴 처리시간

'복사 후 전송' 방식은 액티브 노드의 개수가 증가하여도 전송지연시간이 많이 길어지지 않는 반면, 기존의 방식은 전송지연시간이 크게 증가하는 것을 볼 수 있다. 또한 트래픽의 양이 증가할수록 두 방식의 전송지연시간의 차이가 커지는 것을 알 수 있다. 기존의 방식은 액티브 패킷이 거치는 중간 액티브 노드에서 프로그램 실행을 위한 전체 패킷이 모두 전송될 때까지 대기 상태에 있게 되기 때문에 중간 액티브 노드의 수가 증가할수록, 트래픽의 양이 증가할수록 전송지연시간이 길어지게 되는 것이다. 따라서 액티브 네트워크에서 중간 액티브 노드에서 패킷을 변화시키지 않을 경우에는 '복사 후 전송' 방식을 사용하는 것이 보다 효율적이다.

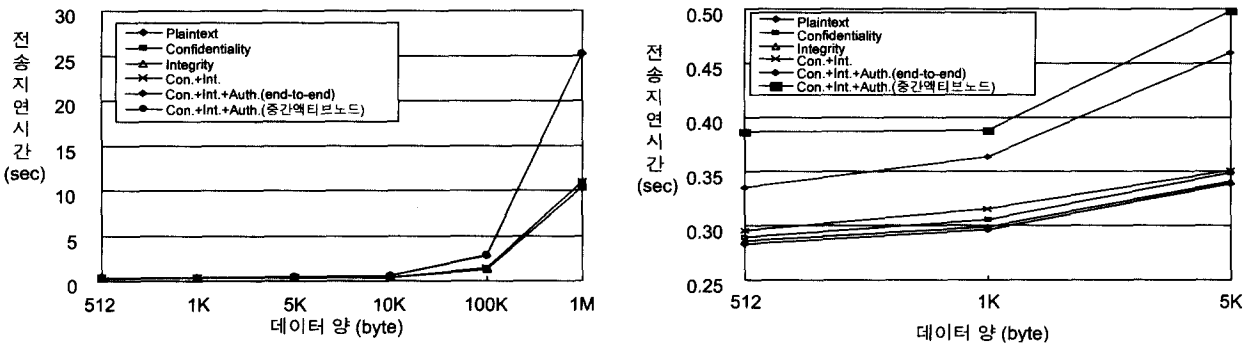
- 보안 기법을 적용할 때

송신자와 수신자, 중간의 액티브 노드에서 다양한 보안 기법을 적용하였을 때의 전송지연시간을 측정하였다. 기밀성 서비스를 위해서는 비밀키 기반의 암호 알고리즘인 DES(Data Encryption Standard)의 CBC Mode(Cipher Block Chaining Mode)를 사용하고, 무결성 서비스를 위해서는 MD5를 이용하였다. 송신자 혹은 중간 노드를 인증하면서 기밀성과 무결성을 제공하기 위해서는 공개키 기반의 암호 알고리즘인 RSA(Rivest-Shamir-Adleman)을 이용하였다. (그림 7)은 보안 기법을 적용하면서 액티브 패킷을 임의의 5개의 중간 액티브 노드를 거치며 전송하였을 때의 전송지연시간을 나타낸다. 중간 노드간에는 최소의 홉 수가 되도록 송신 경로를 지정하였다. 보안을 위한 처리 시간 때문에 전송하는 데이터의 양이 증가할수록 보안 강도가 강해질수록 전송지연시간이 길어지는 것을 볼 수 있다.

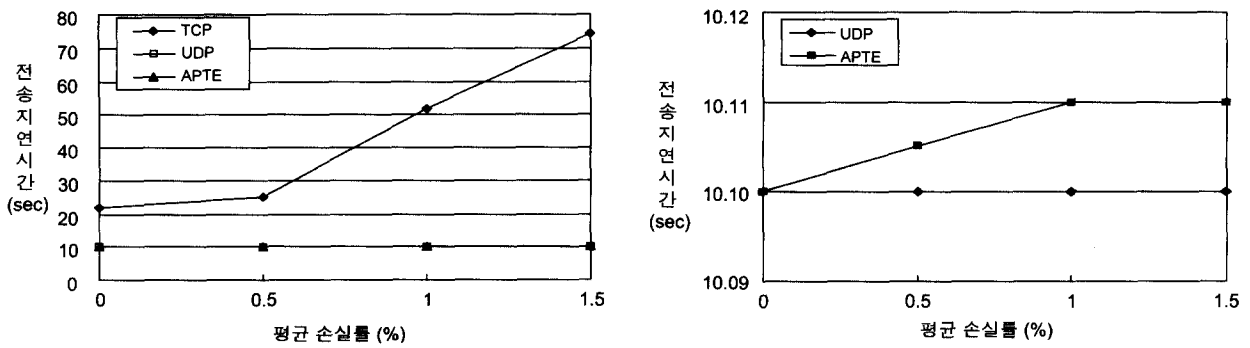
데이터가 전송 중에 위·변조되지 않았음은 송신지에서 메시지의 해쉬값을 생성하고 메시지와 같이 전송하면 수신지가 검증을 통하여 입증할 수 있다. 해쉬값을 계산하는 시간은 비교적 작기 때문에 평문 전송 시간과 크게 차이가 없다. 따라서 무결성 서비스는 큰 오버헤드 없이 제공할 수 있다. 기밀성 서비스는 암·복호화 프로세싱으로 인해 평문 전송에 비해 전송시간의 지연이 크고, 데이터의 양이 증가할수록 암·복호화 시간도 증가하므로 평문과의 전송시간의 차이가 커진다. 데이터가 정당한 사용자로부터 전송되었음을 증명하는 인증 서비스를 제공함으로써 보다 강력한 보안을 제공할 수 있으나 평문과 무결성, 기밀성만을 제공하는 것에 비하여 많은 오버헤드를 생기는 것을 알 수 있다. 그리고 종단간의 인증만을 제공하는 경우에 비해 중간 노드의 인증을 적용하는 경우에 더 큰 오버헤드가 발생한다. 따라서 인증 서비스는 반드시 필요한 경우에 적절히 사용할 필요가 있다.

(2) 백그라운드 트래픽이 있을 때

(그림 8)은 패킷 손실이 일어났을 때의 전송지연시간을 백그라운드 트래픽을 점점 증가시키면서 측정한 결과이다. 시뮬레이션은 모든 프로토콜이 반드시 거치게 되는 자율 시스템의 경계 라우터(AS Boundary Router)와 자율 시스



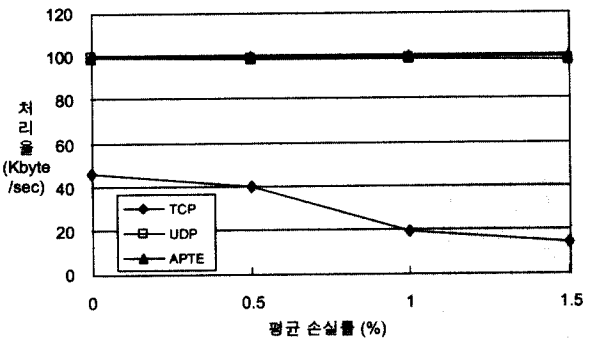
(그림 7) 데이터 양에 따른 보안 강도 별 전송지연시간



(그림 8) 백그라운드 트래픽이 있을 때의 전송지연시간

템 내 영역의 경계 라우터(Area Border Router)를 액티브 노드로 설정하고 실행하였다. UDP는 패킷 손실에 대하여 어떤 대응도 하지 않고 TCP는 손실의 원인을 혼잡으로 간주하고 전송률을 줄인다. 반면에 제안한 APTE는 전송률은 변화시키지 않고 수신 측에서 손실을 감지하면 손실된 액티브 패킷에 대한 재전송을 송신자 혹은 경로 상의 이전 노드에게 요청한다. 따라서 손실을 복구하면서도 전송지연 시간에는 큰 영향을 끼치지 않는다. (그림 8)(a)는 손실률이 증가할수록 TCP는 전송지연시간이 급격히 늘어나고 있지만 APTE의 전송지연시간은 손실에 어떤 대응도 하지 않는 UDP와 별 차이가 없다는 것을 보여준다. (그림 8)(b)를 보면 손실률이 증가할수록 APTE의 전송지연시간이 길어진다. 이는 액티브 패킷을 전송하면서 복구가 일어나고 있기 때문이다. 하지만 UDP와 크게 차이가 나지 않음을 알 수 있다.

(그림 9)는 백그라운드 트래픽이 있을 때 최종 수신자가 초당 받는 데이터 량을 측정한 결과이다. 이 그래프는 UDP와 큰 차이가 나지는 않지만 APTE의 처리율이 가장 높다는 것을 보여주고 있다. TCP는 손실을 감지할 때마다 전송률을 감소시키기 때문에 처리율이 가장 낮다. UDP는 전송률의 변화 없이 데이터를 일정하게 보내지만 손실에 대한 복구 능력이 없기 때문에 수신자가 받는 데이터의 양이 줄어들게 된다. 따라서 전송 시간이 크게 변하지 않고 손실된 패킷을 모두 복구하는 APTE의 처리율이 가장 높다.



(그림 9) 백그라운드 트래픽이 있을 때의 처리율

5. 결 론

본 논문에서는 IP 망 내의 액티브 노드의 가입과 탈퇴 여부, 위치, 실행 환경에 대해 발견하는 기법을 적용하고 액티브 패킷들을 손실 없이 빠르고 안전하게 실행을 원하는 노드로 전송하는 기법을 제안하였다.

액티브 노드의 발견을 위해서는 OSPF의 Opaque LSA를 이용하는 방법을 적용하였다. 액티브 노드는 자신에 대한 정보를 LSA를 이용하여 링크 상태 업데이트 패킷을 플러딩함으로써 네트워크 내의 다른 액티브 노드들에게 알린다. 이 방법은 완전히 동적으로 동작하므로 네트워크의 동적인 변화를 인식시키지 못하고 관리자의 노력을 많이 요구하는 정적인 네트워크 토폴로지의 구성으로 인한 단점을 극복할 수 있다. 뿐만 아니라 기존의 중앙 집중화된 방식을 이용하는 경우에 하나의 노드의 실패로 인하여 네트워크가 마비될 수도 있는 위험에 대한 부담도 가지지 않는다.

액티브 노드로 손실 없이 액티브 패킷을 전달하기 위해서는 수신자가 손실을 감지하여 송신자에게 재전송을 요구하도록 하는 신뢰성 전송 방식을 제안하였다. 수신자가 일련번호를 이용하여 들어오는 패킷의 신뢰성 검사를 하고 손실을 감지하는 대로 재전송을 요구하는 방식은 패킷의 전송과 함께 손실 복구가 이루어지므로 복구로 인한 지연이 적다는 장점이 있다. 또한 액티브 패킷의 효율적인 전송을 위해서는 '복사 후 전송' 방식을 이용하여 전송지연을 줄였다. 액티브 노드와 액티브 패킷에 안전성을 제공하기 위해서는 선택적인 보안 메커니즘을 적용할 수 있도록 하였다.

그리고 이러한 전송 기법을 제공하는 액티브 패킷 전송 엔진을 제안하였다. 액티브 패킷 전송 엔진은 UDP와 ANEP 사이에서 액티브 패킷을 처리하고 사용자 레벨에서 구현이 가능하므로 기존의 프로토콜의 변경 없이 쉽게 액티브 노드에 적용할 수 있다. 또한 IP 패킷의 흐름에도 영향을 끼치지 않는다. 뿐만 아니라 실행 환경보다 하위 계층에서 액티브 패킷에 보안 기법을 적용함으로써 보안의 강도는 높이고, 모든 응용들이 신뢰성과 효율성 및 안전성을 위한 기능을 별도로 가지지 않아도 되도록하여 시스템 자원의 낭비를 막는다.

적용한 액티브 노드 발견 기법과 액티브 패킷 전송 엔진의 성능을 평가하기 위하여 기존의 라우터와 액티브 라우터가 공존하는 IP 망의 환경을 구성하여 시뮬레이션을 수행하였다. 적용한 액티브 노드 발견 기법의 성능을 중앙 집중화된 방식과 비교한 결과, 가입과 탈퇴 시의 지연이 모두 적었다. 또 TCP, UDP를 비교 모델로 설정하고 제안한 액티브 패킷 전송 엔진을 시뮬레이션한 결과를 통하여 제안한 액티브 패킷 엔진이 신뢰성 전송을 보장하면서도 어떤 기능도 제공하지 않는 UDP에 비하여 부하가 크지 않고 성능이 뛰어나다는 것을 입증할 수 있었다.

IP 망에 액티브 네트워크의 기술을 도입함으로써 종단에서만 수행하던 기존의 네트워크의 기능을 분산시키고 사용자의 망에 대한 요구 기능을 시기 적절하게 반영할 수 있다. 네트워크 내에서 물리적으로 직접 연결되지 않은 액티브 노드들이 동적으로 토폴로지 정보를 구성하고 관리하는 액티브 노드 발견 기법과 하나의 프로그램이 여러 개의 패킷으로 분할되어 전송되더라도 신뢰성을 보장하여 정상적인 작업을 수행할 수 있도록 하는 액티브 패킷의 신뢰성 전송 기법을 제안하고 시뮬레이션을 통하여 효율성을 증명한 본 논문의 결과는 추후 액티브 네트워크 기술의 IP 망으로의 도입에 기본 자료로서 활용될 수 있을 것이다.

본 논문에서는 적용한 액티브 노드 가입, 탈퇴에 대한 발견 기법은 자율 시스템 내에서만 동작할 수 있다. 따라서 자율 시스템간의 액티브 노드 토폴로지 정보 유지에 관한 연구가 수행되어야 한다. 또한 중간 액티브 노드에서 패킷의 일부만을 수정하고 그 부분만에 대한 책임을 지는 경우와 같은 다양한 상황에 적절한 보안 메커니즘의 적용에 대한 연구도 수행되어야 할 것이다.

참 고 문 헌

[1] D. L. Tennenhouse, J. M. Smith, W. D. Sncoskie, D. J. Wetherall, and G. J. Minden, "A Survey of Active Network Research," IEEE Communications Magazine, Vol.35, No.1, pp.80-86, 1997.

[2] Pankaj Kakkar, "The Specification of PLAN," Jul., 1999.

[3] Jonathan T. Moore and Michael Hick, "A Service Layer Routing Protocol for PLAN," Nov., 1997.

[4] R. Coltun, "The OSPF Opaque LSA Option," IETF RFC 2370, 1998.

[5] D. Galand, O. Marce, "Active Router Information in Routing Protocols," IETF Internet Draft, 2000.

[6] Active Networks Assigned Number Authority, http://www.isi.edu/~braden/anana.

[7] J. Moy, "OSPF Version 2," IETF RFC 2328, 1998.

[8] Li-wei H. Lehman, Stephen J. Garland, and David L. Tennenhouse, "Active Reliable Multicast," IEEE INFOCOM '98, 1998.

[9] Sneha Kumar Kasera, Supratik Bhattacharyya, Mark Keaton, Diane Kiwior, Jim Kurose, Don Towsley and Steve Zabele, "Scalable Fair Reliable Multicast Using Active Services," IEEE Network Magazine(Special issue on Multicast), 2000.

[10] Internet Assigned Numbers Authority, http://www.iana.org

[11] D. Scott Alexander, Bob Braden, Carl A. Gunter, Alden W. Jackson, Angelos D. Keromytis, Gary J. Minden, and David Wetherall, "Active Network Encapsulation Protocol (ANEP)," http://www.cis.upenn.edu/switchware/ANEP/docs/ANEP.txt, 1997.

[12] 윤보영, 채기준, 이수형, 남택용, "액티브 노드에서의 액티브 패킷 실행을 위한 경로 설정 및 신뢰성 전송기법", 제12회 통신정보합동학술대회(JCCI) 논문집, Apr., 2002.

[13] Network Simulator ns-2,. http://www.isi.edu/nsnam/ns

[14] Crypto++TM Library 5.0, http://www.eskimo.com/~weidai/cryptlib.html.

[15] Kenneth L. Calvert, Matthew B. Doar, Ellen W. Zegura. "Modeling Internet Topology," IEEE Communications Magazine, Jun., 1997.

[16] Christoph Hanle, Markus Hofmann, "Performance Comparison of Reliable Multicast Protocols using the Network Simulator ns-2," IEEE the Proceedings of the Annual Conference on Local Computer Networks(LCN), Oct., 1998.

[17] Kevin Thompson, Gregory J. Miller, Rick Wilder, "Wide-Area Internet Traffic Patterns and Characteristics," IEEE Network, Nov./Dec., 1997.

김 방 은

e-mail : bangeun.kim@samsung.com
2000년 이화여자대학교 컴퓨터학과 (공학사)
2003년 이화여자대학교 컴퓨터학과 (공학석사)
2003년~현재 삼성전자(주) 네트워크 사업부 NMS Lab. 연구원
관심분야 : 네트워크 관리 및 보안, 인터넷/무선통신망 프로토콜 설계 및 성능분석

채 기 준

e-mail : kjchae@ewha.ac.kr
1982년 연세대학교 수학과(이학사)
1984년 미국 Syracuse University 컴퓨터학과(이학석사)
1990년 미국 North Carolina State University 컴퓨터공학과(공학박사)
1990년~1992년 미국 해군사관학교 컴퓨터학과 조교수
1992년~현재 이화여자대학교 컴퓨터학과 교수
관심분야 : 네트워크 보안, 인터넷/무선통신망/고속통신망 프로토콜 설계 및 성능분석

김 동 영

e-mail : dykim@etri.re.kr
1993년 건국대학교 전산학과(공학사)
1998년 건국대학교 대학원 전산학과 (공학석사)
1998년~2001년 (주)윈베이스 소프트웨어
2001년~현재 한국전자통신연구원 정보보호연구단 연구원
관심분야 : 네트워크 보안, 액티브 네트워크, 프로그래밍 언어

나 중 찬

e-mail : njc@etri.re.kr
1986년 충남대학교 계산통계학과(이학사)
1989년 숭실대학교 대학원 전산학과 (공학석사)
2004년 충남대학교 대학원 컴퓨터과학과 (이학박사)
1989년~현재 한국전자통신연구원 정보보호연구단 책임연구원
관심분야 : 네트워크 보안, 실시간 시스템, 고속통신망 성능분석