

리눅스 상에서 ATMARP 다중 서버 구현

정희원 서은미*, 박광로**, 장일순*, 조경록*, 유영갑*

An Implementation of a ATMARP Multiple Server on the LINUX

Eunmi Seo*, Kwangroh Park**, Ilsoon Jang*, Kyoungrok Cho*, Younggap You*

Regular Members

요 약

수년 전부터 리눅스 운영체제 환경에서 ATM을 지원하기 위한 연구가 활발히 진행되어 왔다. 기존의 리눅스 환경에서는 하나의 LIS에 하나의 ATMARP 서버가 독립적으로 존재하는 ATMARP 단일 서버가 설계되었으나, 본 논문에서는 리눅스 기반의 IPOA 상에서 하나의 ATMARP 서버가 다수 개의 LIS를 관리할 수 있도록 ATMARP 다중 서버를 구현하였다. ATMARP 다중 서버를 구현하기 위해서는 ATMARP 프로토콜 처리 기능과 관리 테이블의 추가 및 삭제 기능이 필요하다. 이러한 기능이 수행됨으로써 ATMARP 단일 서버 환경에서 ATMARP 서버의 수가 증가되는 문제점을 해결하였으며, LIS간의 연결 설정 및 파일 전송 실험을 통해 다수의 LIS가 하나의 ATMARP 서버로 관리됨을 보였다. 그 결과 ATMARP 단일 서버 환경에서 보다 ATMARP 서버의 수를 감소시킬 수 있었으며 더욱 더 향상된 성능의 망 구축을 가능하게 하였다.

ABSTRACT

Recently, supporting ATM networks under LINUX operating systems environment has been actively studied. An ATMARP server in a conventional LINUX environment was designed under the ATMARP single-server to operate independently within a Local IP Subnet(LIS). In this paper, an ATMARP multi-server is introduced and implemented an ATMARP-server managing multiple LIS's on the LINUX-based IPOA. The ATMARP protocol processing functions and addition/deletion functions on management tables are required to implement the ATMARP multi-server. These functions can solve the problem of increasing the number of ATMARP servers in ATMARP single-server environment. The file transmission experiment with a connection configuration between LIS's shows that multiple LIS's can be managed by an ATMARP-server. As the results, the ATMARP multi-server scheme yields a smaller number of servers than that of the conventional ATMARP single-server scheme, and the network composition can be improved substantially.

I. 서론

ATM(Asynchronous Transfer Mode)은 WAN(Wide Area Network)에서 뿐만 아니라, LAN(Local Area Network)간의 접속을 위해 고속의 백본 망에서 이용되는 가장 일반적인 기술이다. 따라

서B-ISDN(Broadband Integrated Services Digital Network)과 같은 고속의 통신망에서 LAN과 같은 비 연결형 서비스를 지원하는 것이 무엇보다도 필요하다. 현재 인터넷에서 사용되고 있는 TCP(Transmission Control Protocol)는 연결형 서비스이고, IP(Internet Protocol)는 비 연결형 서비스이므로 TCP/IP 응용 서비스들을 연결형 망에 적용시키려면

* 충북대학교 정보통신공학과 (emseo@hbt.chungbuk.ac.kr)

** 한국전자통신연구원 교환전송기술연구소

논문번호:99394-0928, 접수일자 : 1999년 9월 28일

ATM 망의 연결형 기술 위에 비 연결형 서비스가 제공되어야 한다. 이를 위해 IETF(Internet Engineering Task Force)에서 Classical IP over ATM (IPOA)이 제안되었으며, 이는 ATM 망을 이용해서 비 연결형 서비스를 제공하는 가장 일반적인 기술 중의 하나이다^[1].

기존의 IPOA 기반의 ATM 망은 RFC1577(Requests For Comments 1577)에 따라 각각의 LIS(Logical IP Subnet)내에 ATMARP(ATM Address Resolution Protocol)서버와 클라이언트가 존재하도록 구성되어 있다. 이를 위해 ATMARP 서버가 클라이언트의 ATM 주소와 IP 주소를 등록 받아 저장하게 된다. 이로써 클라이언트에게 IP 주소에 대응되는 ATM 주소를 제공하게 되며, 클라이언트는 ATMARP 서버에 의해 분석된 ATM 주소로 상호간의 데이터 전송을 수행하게 된다. IPOA 상에서의 이러한 수행 과정은 IP 패킷이 LLC/SNAP(Logical Link Control/Sub Network Access Protocol) Encapsulation을 이용한 AAL5(ATM Adaptation Layer 5) 패킷으로 캡슐화된 후 가능해진다^[2-4].

그러나 일반적인 ATM 망은 다수의 LIS로 구성되고, 라우터를 통해 다른 LIS내 클라이언트와의 연결을 설정한다. 이러한 ATM 망에서 기존의 IPOA는 각각의 LIS마다 ATMARP 서버가 독립적으로 존재하여야 함으로 하나의 ATMARP 서버가 하나의 LIS에 대한 ATM 주소 정보만을 제공하도록 되어있다. 따라서 LIS가 증가되면 ATMARP 서버의 수도 증가되기 때문에 간소화된 망의 구성이 불가능해지고, 그 결과 ATM 망은 성능이 저하될 수밖에 없고 확장성 측면에서도 많은 문제가 있다.

따라서 리눅스 기반의 IPOA 상에서 ATMARP 단일 서버 환경에서 망 성능이 저하되는 문제점을 해결하고자 ATMARP 다중 서버를 제안하고^[5] 구현하였다. ATMARP 다중 서버의 동작은 기존의 RFC1577과 최근 표준안으로 확정된 RFC2225를 따른다. 다중 서버는 확장된 ATMARP 프로토콜 처리 기능과 관리 테이블에 속하는 LIS 테이블·영구 테이블의 추가 및 삭제 기능을 수행하도록 하였다. ATMARP 다중 서버를 구현 검증하기 위한 실험 환경으로 ATMARP 다중 서버와 ATMARP 단일 서버로 ATM 망을 구성하였으며, ATMARP 서버와 클라이언트간은 VC로 연결되도록 하였다. 두 서버 환경에서는 ATMARP 서버와 클라이언트간의 연결 여부를 판별하기 위해 TTCP(Test TCP)를 이용한

IP 라우팅 실험을 수행하였고, 설정된 VC 상에서 파일 전송을 통한 ATMARP 서버의 성능을 분석하였다.

본 논문의 2장은 기존의 ATMARP 단일 서버 환경에서 구성되는 Classical IP over ATM의 일반에 대해 기술하였다. 3장에서는 ATMARP 다중 서버의 요구 사항과 ATMARP 서버에 추가된 기능에 대해 기술하였다. 4장에서는 ATMARP 다중 서버의 구현을 확인하기 위해 IP 라우팅 실험과 ATMARP 서버의 성능 실험을 수행하였으며, 실험을 통한 분석 결과를 제시하였다. 마지막으로 5장에서는 분석 결과를 토대로 결론을 맺고자 한다.

II. Classical IP over ATM

IPOA는 RFC1577에 따라 ATM 상에서 IP 패킷을 전달하기 위한 방법으로 제안되었다. 이는 IP 패킷을 LLC/SNAP Encapsulation을 이용하여 AAL5 패킷으로 캡슐화하고, 캡슐화된 셀은 ATM 종단 시스템간에 ATM 연결을 통하여 통신한다. 이러한 IP 패킷을 전송하기 위해서는 IP 패킷의 IP 주소를 ATM 주소로 분석하고, 시그널링을 통해 설정된 VC(Virtual Channel) 상에서 수행된다^[6-7]. 이러한 VC는 시그널링을 통해 연결을 설정하는 SVC(Switched Virtual Connection)와 시그널링을 하지 않고 영구적 연결을 설정하는 PVC(Permanent Virtual Connection)로 구분된다. 즉 IPOA에서는 SVC를 지원하기 위해 ATM 신호와 IP 주소를 ATM 주소나 VC 식별자로 매핑시키기 위해 필요한 ATMARP(ATM Address Resolution Protocol) 주소해석 기능을 지원해야 한다.

일반적으로 ATM 망은 VC로 연결된 다수의 LIS로 구성되어 있으며, 기존의 IPOA는 각각의 LIS마다 ATMARP 서버가 존재해야 하는 ATMARP 단일 서버로 구성되어 있다. ATMARP 단일 서버 환경에서는 망 내 존재하는 모든 LIS가 IP 네트워크 주소로 구분되며, 라우팅에 의해 LIS간의 통신이 가능해진다. 따라서 ATMARP 서버와 클라이언트는 하나의 LIS 내에 존재할 경우 동일한 IP 네트워크 주소를 가지게 되고, 이로써 ATMARP 서버와 클라이언트가 ATM 망에 직접 연결될 수 있다. IPOA 상에서 주소 해석을 위해 ATMARP(ATM Address Resolution Protocol)와 InATMARP(Inverse ATMARP)의 두 메커니즘이 사용되며, 이러한 두

메커니즘으로 연결 설정과 ATMARP 기능이 가능해진다. ATMARP 단일 서버 환경에서의 LIS 요구 사항은 다음과 같다.

- 하나의 LIS내 모든 멤버는 동일한 IP 네트워크/서버넷 주소 및 네트워크 마스크 유지
- 하나의 LIS내 모든 멤버는 ATM 망에 직접 연결되어야 하고, LIS내 모든 멤버는 같은 LIS의 다른 멤버와 ATM을 통해 통신
- 하나의 LIS내 모든 멤버는 다른 LIS의 멤버와 IP 라우트를 통해 연결
- LIS의 모든 멤버는 PVC를 사용할 때, InATMARP를 통해서 VC를 IP 주소로 매핑
- LIS의 모든 멤버는 SVC를 사용할 때, ATMARP를 통해서 IP 주소를 ATM주소로, InATMARP를 통해서 ATM주소를 IP주소로 매핑
- RFC1577에서 정의된 기본적인 IP 네트워크를 구현
- ATM Forum에서 정의한 ATM 주소를 지원
- IEEE 802.2 LLC/SNAP Encapsulation을 지원
- 기본 MTU(Maximum Transmission Unit) 크기는 9188 옥텟(9180+8(LLC/SNAP 헤더))
- 인코딩은 AAL5를 지원

그러나 기존의 IPOA에서는 각 LIS(Logical IP Subnet)에서 하나의 ATMARP 서버만을 고려하여 설계됨으로써 여러 개의 LIS망으로 구성되는 일반적인 ATM 망 환경에서 망의 구성을 복잡하게 하고 망 구성의 유연성 및 확장성에 제약 받는 문제점이 있다. 또한 LIS간에 라우팅 기능을 통해 연결함으로써 망 성능을 저하시키는 문제점이 있다.

III. ATMARP 다중 서버

ATM 기반 ATMARP 다중 서버는 스위치와 라우터가 통합된 미래의 네트워크 형태에서 사용자에게 저렴한 가격과 서비스 품질(QoS)을 보장할 수 있는 scalable한 대역폭을 제공할 뿐 아니라, 기존의 LAN과 WAN 기술을 통합할 수 있는 기술이다. 특히 ATMARP 다중 서버는 ATM 교환기의 UNI(User Network Interface)에 접속되어 LIS내의 클라이언트들이 IP 주소에 대응된 ATM 주소를 제공하는 역할을 수행하며 LIS간 트래픽을 전달해 주는 라우터의 역할을 선택적으로 수행한다.

3.1 요구사항

ATMARP 다중 서버는 기존의 ATMARP 단일 서버 환경에서 ATMARP 서버의 수를 감소시키기 위해 하나의 ATMARP 서버가 다수의 LIS를 관리할 수 있도록 하였다. 이는 하나의 ATMARP 서버를 논리적으로 구성함으로써 가능해진다. 따라서 하나의 ATMARP 서버가 다수의 LIS 내 존재하는 클라이언트의 주소를 분석할 수 있게 되었고, 일반적인 ATM 망에서 ATMARP 서버의 수를 줄일 수 있다. ATMARP 다중 서버 환경에서의 LIS 요구 사항은 다음과 같다.

- ATM Forum의 AESA(ATM End System Address) 및 Embedded E.164 주소 체계 지원 가능
- ATMARP 서버는 IPOA 클라이언트들과 PVC 또는 SVC로 접속 가능
- LIS당 128개 이상의 가상연결 설정이 가능하며, 하나의 ATM 인터페이스에는 1024개 이상의 가상연결 설정 가능
- 하나의 ATM 인터페이스에 16개 이상의 논리적 ATMARP 서버 구성 가능
- 각각의 논리적 ATMARP 서버는 주소체계에 의해 구분 가능
- 각각의 논리적 ATMARP 서버 생성 시 selector value, IP 주소 및 서버넷 마스크 값을 입력하도록 구성
- 각각의 논리적 ATMARP 서버는 클라이언트의 기능을 동시에 수행 가능
- ATMARP 서버는 RFC1577 및 RFC2225 동시지원 중 선택적 사용 가능
- 클라이언트들은 LIS간에 고속의 IP 라우팅 가능
- ATMARP 서버 내 테이블이나 엔트리의 값을 수동으로 추가/삭제/변경 가능
- ATMARP 서버는 control connection과 data connection의 독립적 설정이 가능
- control/data connection 각각은 UBR 또는 VBR(PCR, SCR) 선택 가능

3.2 ATMARP 다중 서버의 구성

ATMARP 다중 서버 기능을 지원하기 위해서는 하나의 서버가 여러 개의 LIS를 관리할 수 있어야 한다. 이를 위해 서버는 LIS 테이블을 관리하는 기능과 클라이언트가 시그널링만이 존재하는 단말일 경우를 고려해 영구적인 ATMARP 테이블을 관리

할 수 있는 기능을 가져야 한다. ATMARP 다중 서버 환경에서는 LIS를 관리하기 위한 방법으로 IP 네트워크 주소뿐만 아니라, ATM 주소도 구분해서 관리해야 하며, 이를 위해 LIS마다 ATM 주소 필드에 존재하는 한 바이트 SEL(selector)값을 이용하였다. 따라서 각 클라이언트에서는 ATMARP 서버의 ATM 주소를 설정할 때 반드시 SEL 값을 고려해야 한다.

리눅스 환경에서 ATMARP 다중 서버를 구성하기 위해 필요한 소프트웨어 구성은 그림 1과 같다. 리눅스 ATM 환경에서 현재 디렉토리가 ATM일 때의 부 디렉토리는 다음과 같다.

- sigd는 UNI 3.0 및 UNI 3.1 시그널링 데몬 디렉토리
- saal은 시그널링 관련된 AAL 디렉토리
- qgen은 Q.2931 형태의 시그널링 메시지를 처리하는 디렉토리
- arpcltd는 ATMARP 다중 서버 기능을 수행하기 위한 클라이언트 데몬 디렉토리
- arpsrvd는 ATMARP 다중 서버 기능을 수행하기 위한 서버 데몬 디렉토리
- ilmid는 ilmi 주소 등록 데몬 디렉토리
- maint는 ATM을 관리하기 위해 필요한 프로그램을 수행하기 위한 디렉토리
- ip는 atmarp를 수행하기 위한 디렉토리
- lib는 응용 프로그램이나 데몬에서 필요한 라이브러리를 저장하는 디렉토리

ATMARP 다중 서버 기능을 수행하기 위한 기능 블록은 LIS 관리부, 영구테이블 관리부 및 ATMARP 프로토콜 처리부로 구성된다. 이들 각 기능 블록은 리눅스 ATM 환경의 소프트웨어 기능과 정합하여 다중 서버 기능을 수행하게 된다. 그림 2는 다중 서버를 구성하기 위한 ATMARP 서버의 기능 블록 구성도 이다.

이제 이들 구성 블록에 대하여 설명한다. 먼저 LIS 관리부에서는 서버 관리자로부터 입력되는 명령어의 형태에 따라 LIS 테이블을 추가, 삭제 및 현재 테이블의 상황을 출력하는 기능을 수행하며, 다중 서버 기능을 지원하기 위해 LIS 테이블을 관리한다. 따라서 LIS 테이블 관리는 SEL 값으로 구분된 다수의 LIS가 하나의 ATMARP 서버로 관리되도록 한다.

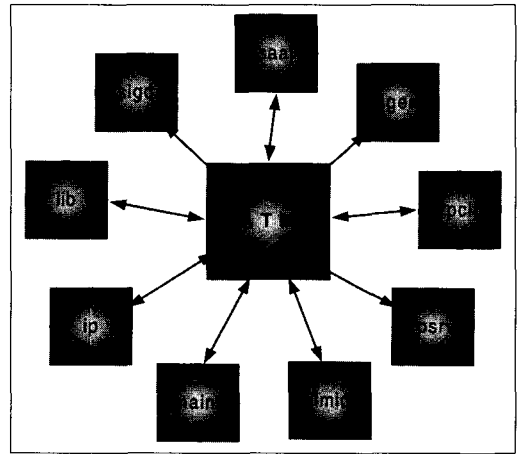


그림 1. ATMARP 다중 서버를 구성하기 위한 소프트웨어 구성

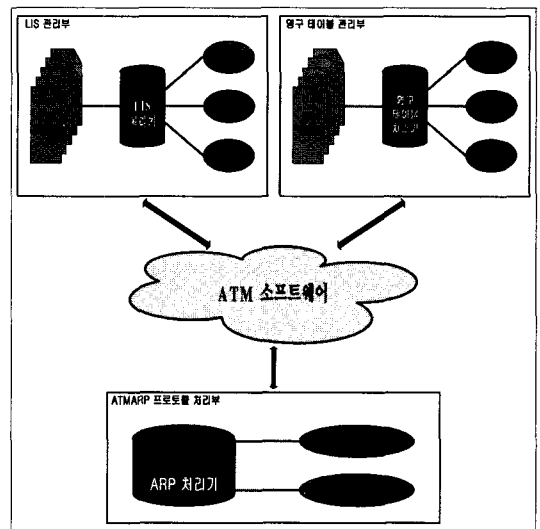


그림 2. ATMARP 다중 서버의 기능블럭 구성도

영구테이블 관리부에서는 서버 관리자로부터 입력되는 명령어의 형태에 따라 영구 테이블을 추가, 삭제 및 출력하는 기능을 수행하며, 다중 서버 기능을 지원하기 위해 영구 테이블을 관리, 운용한다. 이는 특정 클라이언트가 서버로 주소 등록이 불가능할 때 즉 주소 설정 기능이 없이 단지 시그널링 기능만을 가진 클라이언트에도 다중 서버 기능을 지원하기 위한 것이다.

ATMARP 프로토콜 처리부에서는 LIS 관리부와 영구 테이블 관리부와 정합하여 클라이언트와 다중 서버간에 주소 해석 기능을 수행하며, 다중 서버 기능을 지원하기 위해 ATMARP 테이블을 관리 및

운용한다. 즉, 서버로 등록된 클라이언트의 주소관리 기능과 함께 주소 해석 요청에 따라 클라이언트에게 통화하고자 하는 상대방 ATM 주소를 해석해 통보하는 기능을 수행한다.

3.3 ATMARP 다중 서버의 동작

ATMARP 프로토콜을 처리하기 위해서는 하나의 ATMARP 서버가 다수 개의 LIS를 관리할 수 있도록 ATM 망을 구성해야 한다. 이때 상호간의 연결을 위해 ATMARP 서버와 클라이언트간에 VC가 설정되어야 한다. 설정된 VC 상에서는 ATMARP 요청과 응답을 통한 ATMARP 프로토콜 처리 과정이 수행된다. 이러한 과정으로 IPOA 상에서 유연성이 보장되는 확장된 ATM 망의 구성이 가능해진다. 그림 3은 ATMARP 다중 서버 환경에서 클라이언트와 ATMARP 서버가 ATMARP 프로토콜을 처리하는 흐름도이다.

ATMARP 다중 서버의 동작 단계는 크게 테이블 설정, 주소 등록 설정, 주소 해석 요청 및 주소 해석 통보단계로 구분할 수 있다. 첫 번째 테이블 설정 단계를 보기로 하자. 우선 서버 관리자는 단말에서 테이블 설정 명령어를 사용할 수 있다. 테이블 설정 명령어는 다중 서버를 지원하기 위해 LIS 테이블을 추가 및 삭제하는 명령어와 영구 ATMARP 테이블을 추가 및 삭제하기 위한 명령어로 구성되

어 있다. 관리자의 입력 메시지로부터 데이터를 추출하여 해당 테이블 설정 기능을 수행하고, 상기의 과정을 수행하는 동안 오류가 있는 경우에는 관리자에게 오류 메시지 코드를 포함하는 오류 메시지를 통보하고 수행을 마친다. 테이블 상에 존재하는 모든 LIS는 반드시 파일로 저장되며, 테이블의 추가 및 삭제 상황은 서버 관리자에 의한 "arpsrv" 명령으로 확인이 가능하다. 그림4는 ATMARP 다중 서버 환경에서 테이블 설정 과정의 흐름도이다.

두 번째 주소 등록 설정 단계에 대하여 설명하기로 하자. 기본적으로 클라이언트에서 VCC 설정요구를 받으면 서버는 클라이언트로부터 수신한 입력 데이터로부터 클라이언트 자신의 ATM 주소와 통신하고자 하는 상대방의 ATM 주소를 추출한다. 추출된 ATM 주소의 SEL 필드의 값을 이용하여 LIS 관리기의 LIS 테이블을 조회한다. 조회 결과 LIS 테이블에 등록되어 있으면 주소 등록을 설정한다. 상기의 과정을 수행하는 동안 오류가 있는 경우에는 관리자에게 오류 메시지 코드를 포함하는 오류 메시지를 통보하고 수행을 마친다.

세 번째는 주소 해석 요청 단계로서 상기 주소 등록 설정 단계를 수행한 후 다중 서버 환경에서 상대방 주소 해석을 요청하는 과정이다. 우선 클라이언트에서 응용 프로그램에 의해 상대방 주소 해석을 요구받으면 자신의 ATM 주소와 IP 주소를 추출하고, 그리고 통신하고자 하는 상대의 IP 주소를 추출한다. 착신지의 ATM 주소를 얻기 위하여 추출한 데이터를 포함한 ARP_REQ 메시지를 서버로 전송하고 망관리 기능을 위한 통계 데이터를 수집한다. 상기의 과정에서 서버로부터 수신된 메시지가 자신의 IP가 아니거나 ARP_REQ에 의한 정상적인 응답 메시지가 아닌 경우에는 관리자에게 오류 메시지 코드를 포함하는 오류 메시지를 통보하고 수행을 마친다.

마지막으로 주소 해석 통보 단계는 다중 서버 환경에서 ATMARP 서버가 클라이언트로부터 상대방 주소 해석을 요구받아 이를 통보해 주는 과정이다. 서버가 클라이언트로부터 주소 해석을 요구받으면 클라이언트가 통신하고자 하는 상대측, 즉, 착신지의 IP를 추출하고 추출된 상대 IP가 ATMARP 테이블에 존재하는지 또는 영구적인 ATMARP 테이블에 존재하는지를 판단한다. 따라서 클라이언트가 통신하고자 하는 상대 ATM 주소를 추출하여 ARP_REQ 메시지를 전송하고 망 관리 기능을 위한 통계 데이터를 수집한다. 상기의 과정에서 오류가

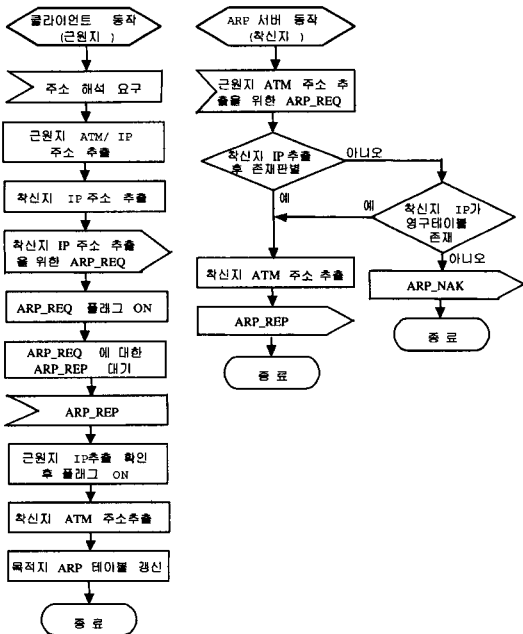


그림 3. ATMARP 프로토콜 처리 흐름도

발생한 경우에는 클라이언트로 ARP_NAK 메시지를 통보하고 수행을 마친다. 이제까지 기술한 ATMARP 다중 서버의 기본적인 동작은 표준안 RFC1577과 RFC2225를 따르며, 표 1에서 이러한 표준안들의 특징을 간단히 비교하였다.

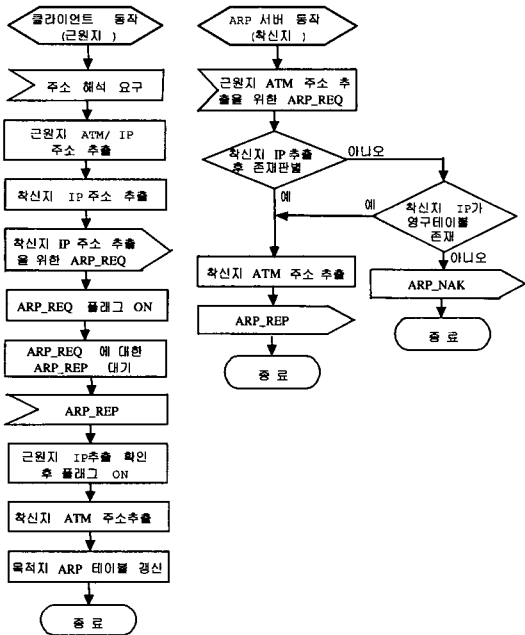


그림 4. ATMARP 프로토콜 처리 흐름도

IV. ATMARP 다중 서버의 구현 실험

리눅스 기반의 IPOA 상에서 ATMARP 다중 서버 기능을 검증하기 위하여 기존의 ATMARP 단일

서버로 구성된 ATM 망과 ATMARP 다중 서버로 구성된 ATM 망을 각각 구성하였다. 이러한 ATM 망에서의 LIS는 ATMARP 서버, 클라이언트 및 라우터로 구성되었으며, LIS 간을 스위치로 연결시킴으로써 상호간의 데이터 전송이 가능하도록 구성하였다. 실험 환경을 구성하기 위해 ATM 카드가 장착된 PC들을 클라이언트 또는 ATMARP 서버로 설정하고 리눅스 커널-2.0.29와 ATM 프로그램인 ATM-0.31을 설치하였다. 단, ATMARP 다중 서버 환경에서는 다중 ATMARP 서버 기능에 적합하도록 확장된 ATM-0.31을 설치하였다. 상기의 과정을 완료한 후 ATM 스위치를 통해 ATMARP 서버와 클라이언트간에 VC를 설정하고 ATMARP 단일 서버 및 ATMARP 다중 서버 각각의 환경에서 IP 라우팅 실험과 ATMARP 서버의 성능 실험을 수행하였다. ATMARP 다중 서버 환경에서 ATMARP 서버와 클라이언트의 정상적인 동작을 보장하기 위해서는 서버 및 클라이언트 관련 데몬 생성, ATM 인터페이스, 라우팅 테이블 상에 설정된 경로 및 해당 LIS의 정상적인 지를 확인해야 한다. 그림 5와 그림 6은 실험을 위해 각각 ATMARP 단일 서버와 ATMARP 다중 서버로 구성된 ATM 망의 구성도이다.

4.1 IP 라우팅 실험

TTCP를 이용하기 위해 ATMARP 서버와 클라이언트를 송신측과 수신측으로 구분하였다. 실험에서는 고정된 버퍼의 길이, 가변적인 버퍼의 수 그리고 고정된 데이터 크기를 성능 파라미터로 사용하였다. IP 라우팅 실험 결과에서 파라미터로 설정된 버퍼의 길이와 수는 서버의 성능을 분석하는 실험에서

표 1. RFC1577과 RFC2225 비교

기능 \ 구분	RFC1577	RFC 2225
클라이언트 등록 기능	InATMARP요청→응답	ATMARP요청→서버로 전달 서버의 테이블 15분에 1번씩 동작 수행
서버 테이블엔트리 에이징시동작	InATMARP요청→응답→엔트리 갱신/제거	해당 엔트리 제거
중복등록감지기능	해당사항 없음	서버에 중복된 엔트리가 발견→엔터티로 통보
ATMARP 서버 선택 기능	해당 사항 없음 (서버 고장 시 신뢰성 문제발생)	클라이언트가 사용 가능한 ATMARP 서버의 서비스를 받음
클라이언트 테이블의 엔트리 에이징시 동작	SVC, PVC : InATMARP요청→응답→갱신	SVC : ATMARP요청→응답→갱신 PVC : RFC1577과 동일 서버 고장 시 : InATMARP요청→응답→갱신

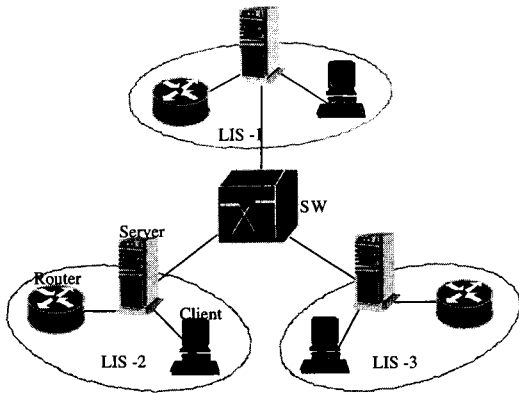


그림 5. ATMARP 단일 서버로 구성된 ATM 망

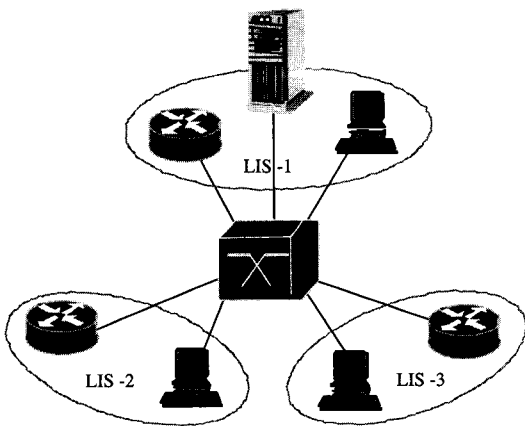


그림 6. ATMARP 다중 서버로 구성된 ATM 망

는 큰 의미를 갖지 못했다. 실험을 통해 ATMARP 단일 서버뿐만 아니라, ATMARP 다중 서버 환경에서도 클라이언트의 주소 분석 기능이 수행될 수 있었다. 또한, ATMARP 다중 서버 환경에서는 하나의 ATMARP 서버로 다른 LIS 내 클라이언트에 대한 ATMARP 기능도 가능해졌다. 이로써, IPOA 상에서 ATMARP 서버와 클라이언트간의 VC 설정이 정상적으로 수행될 수 있었다. 그 결과 송·수신측간의 전송률을 측정할 수 있었으며, 기존의 ATMARP 단일 서버로 구성된 ATM 망이 ATMARP 다중서버로 구성된 ATM 망에서 보다 약간 더 높게 측정되었음을 알 수 있었다. 그림 7은 IP 라우팅 실험을 통한 버퍼 크기별 전송률이다.

4.2 ATMARP 서버의 성능 실험

서로 다른 LIS내에 존재하는 클라이언트간의 데이터를 송, 수신함으로써 실험을 수행하였다. ATMARP 서버의 성능 실험은 망 내 존재하는 모

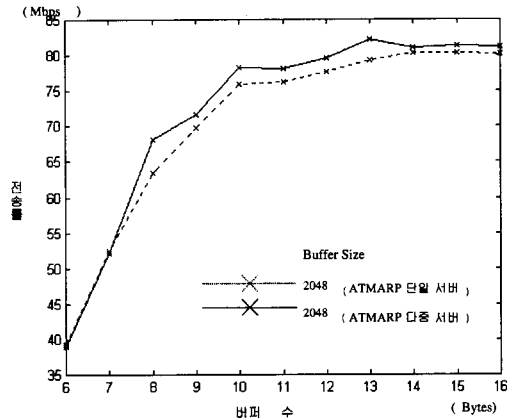


그림 7. 연결 설정 여부 실험

든 클라이언트간을 VC로 연결하고, VC 상에서 송·수신측 간의 파일 전송을 통해 이루어졌다. 기존의 ATMARP 단일 서버 환경에서의 서버는 ATMARP 기능만을 하므로 클라이언트 내의 PC 부하율이 그다지 높지 않게 측정되었다. 그러나 ATMARP 다중 서버 환경에서는 서버가 ATMARP 기능뿐만 아니라 추가된 테이블의 관리와 ATMARP 프로토콜 처리 기능도 수행하기 때문에 PC의 부하율이 다소 높게 측정되었다. 실험을 통해 측정된 전송률은 IP 라우팅 실험 결과에서처럼 기존의 ATMARP 단일 서버 환경에서 좀더 높게 측정되었음을 확인할 수 있었다. 그림 8은 ATMARP 다중서버의 성능 실험 결과이다.

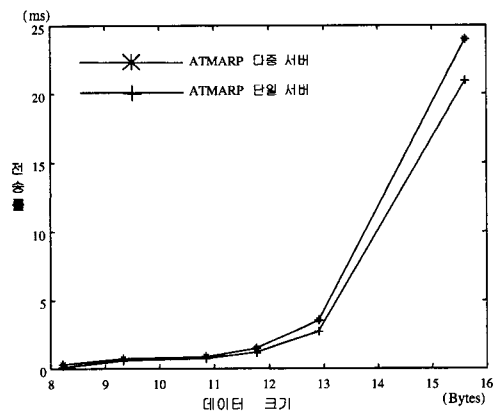


그림 8. ATMARP 다중 서버의 성능 분석

상기의 두 가지 실험 결과 그래프에서는 각각의 서버에 대한 전송률의 차이를 측정할 수 있었으며, ATMARP 다중 서버 환경에서는 ATMARP 단일

서버 환경에 비해 감소된 서버의 수를 확인할 수 있었다. 실험 결과에서의 전송률은 PC의 부하율에 영향을 받게된다. 따라서 기존의 서버에 비해 추가된 기능들이 포함된 ATMARP 다중 서버 환경에서는 PC의 부하율이 증가되었으므로 전송률 또한 낮게 측정되었다. 그러나 두 서버 환경간에 전송률은 극히 미소한 차이였으므로 망 성능에는 큰 영향을 주지 못했다. 일반적인 ATM 망은 라우터를 통해 LIS간의 통신이 가능하므로, 현재 구성된 망 보다 규모가 커질 경우 ATMARP 단일 서버에 비해 월등히 앞서는 ATMARP 다중 서버의 성능을 좀 더 명확히 확인할 수 있을 것이다.

또 다른 결과로는 ATMARP 다중 서버 환경에서 ATMARP 단일 서버에 비해 현저하게 감소된 ATMARP 서버의 수이다. 기존의 IPOA 상에서 감소된 ATMARP 서버의 수는 ATMARP 서버와 클라이언트에 추가된 ATMARP 프로토콜 처리 기능과 관리 테이블의 추가 및 삭제 기능으로 가능해졌다. 따라서 ATMARP 다중 서버로 구성된 IPOA 상에서는 하나의 ATMARP 서버가 다수의 LIS를 관리할 수 있게 됨으로써 소규모의 ATMARP 단일 서버 환경에서 망 성능이 저하되는 문제점을 해결할 수 있었다. 결과적으로 실험에 사용된 소규모 ATM 망은 좀 더 많은 수의 LIS와 라우터로 구성되지 못했기 때문에 서버의 수를 감소시키기는 기능을 갖는 ATMARP 다중 서버의 성능을 충분히 발휘하지 못했다.

V. 결 론

기존의 리눅스 기반의 IPOA는 LIS 마다 ATMARP 서버가 존재하는 ATMARP 단일 서버로 구성되어 있다. 그러나 기존의 ATMARP 단일 서버 환경에서는 LIS가 증가됨에 따라 ATMARP 서버의 수도 증가되어야 하므로 망의 구성이 복잡해진다. 이로 인해 효율적인 데이터 전송이 어렵게 되고, 증가되는 ATMARP 서버의 수로 인해 망 구축 시 경제성과 유연성을 상실하게 된다.

따라서 이 논문에서는 기존의 ATM 망에서 ATMARP 서버의 수가 증가되는 문제점을 해결하고자 하나의 ATMARP 서버를 논리적으로 구성하여 ATMARP 다중 서버를 구현하였다. 다중 서버를 구현하기 위해 확장된 ATMARP 프로토콜 처리 기능과 관리 테이블의 추가 및 삭제 기능을 추가하였

으며, 이로 인해 하나의 ATMARP 서버가 다수 개의 LIS에 대한 주소 분석 기능을 하는 ATM 망의 구성을 가능하게 했다. 그 결과 기존의 ATMARP 단일 서버 환경에 비해 복잡하게 구성되는 망 구조를 단순화할 수 있게 하였으며, 망 구축 시 경제성과 유연성을 가질 수 있게 되어 좀더 효율적인 망의 구성이 가능해졌다. ATMARP 다중 서버의 기능은 향상된 ATM 망을 구성하기 위해 IPOA 상에서만 뿐만 아니라, ATM 기반의 여러 다양한 응용 프로그램들에도 적용될 것이다. 또한 다른 응용 망들과 통합 방안으로 개발되어 좀 더 향상된 망을 구성할 수 있을 것으로 기대된다.

참 고 문 헌

- [1] M. Laubach, *Classical IP and ARP over ATM RFC1577(Request For Comments 1577)*, IETF (Internet Engineering Task Force), Jan. 1994.
- [2] A. Werner, *ATM on Linux*, ftp://lrcftp.epfl.ch, Mar. 1996.
- [3] A. Werner, *The releases of ATM-0.31*, ftp://lrcftp.epfl.ch, Apr. 1997.
- [4] J. Heinanen, *Multiprotocol Encapsulation over ATM Adaptation Layer 5 RFC1483*, IETF, Jul. 1993.
- [5] 박광로, "The design of multiple ARP server on linux," *한국통신학회학술발표회논문집*, 1841~1844쪽, 1998년 8월.
- [6] ITU, "B-ISDN DSS2 User Network Interface Layer 3 specification for basic call/connection control," *ITU-T (International Telecommunications Union-Telecommunication standards) Recommendation Q.2931*, Feb. 1995.
- [7] ITU, "B-ISDN DSS2 User Network Interface Layer 3 specification for point-to-multipoint call/connection control," *ITU-T Recommendation Q.2971*, Oct. 1995.

서 은 미(Eun-mi Seo)

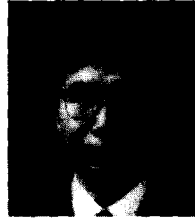
정회원



1997년: 충주 산업대학교 정보통신 공학과 학사.
1998년 3월~현재: 충북대학교 정보통신 공학과 석사과정.
<주관심 분야> ATM 망, 초고속 정보 통신.

유 영 갑(Young-gap You)

정회원



1975년: 서강대학교 전자공학과 공학사.
1981년: 미시간대학교(미국) 전기전산공학과 공학석사.
1986년: 미시간대학교(미국) 전기전산공학과 공학박사.

1988년 3월~현재: 충북대학교 전기전자공학부 교수.
<주관심 분야> Computer architecture, Memory testing, 고속 시스템 설계, HDTV, ATM 등

박 광 로(Kwang-roh Park)

정회원

1982년 2월: 경북대학교 공과대학전자공학과 학사.
1985년 2월: 경북대학교 대학원 석사.
1984년~현재: 한국전자통신연구소 교환전송 기술연구소 무선ATM팀 선임 연구원.

장 일 순(Il-soon Jang)

정회원



1997년: 충북대학교 정보통신 공학과 학사.
1999년: 충북대학교 정보통신 공학과 공학석사.
1999년 3월~현재: 충북대학교 정보통신공학과 박사과정.

<주관심 분야> 디지털 신호처리, 디지털 통신 시스템

조 경 록(Kyoung-rok Cho)

정회원



1977년: 경북대학교 전자공학과 공학사.
1989년: 일본 동경대학교 전자공학과 공학석사.
1992년: 일본 동경대학교 전자공학과 공학박사.

1979년~1986년: (주)금성사 TV연구소 선임 연구원.
1999년~2000년: 오레곤주립대학 객원교수.
1992년~현재: 충북대학교 정보통신공학과 부교수.
<주관심 분야> VLSI 시스템설계, 통신 시스템용 LSI 개발, 고속 마이크로프로세서 설계.