

Linux 기반의 DiffServ over MPLS 망 구축 및 IEEE 1394 활용 방안 연구

Construction of Linux Based DiffServ over MPLS Network for IEEE 1394 Service Provisioning

권호한, 윤상식, 최덕재
전남대학교 정보통신 연구소

{zeany, ssyoon}@tyranno.chonnam.ac.kr, dchoi@chonnam.ac.kr

요 약

오늘날 고대역폭의 전송속도와 QoS(Quality of Service)를 요구하는 실시간 동영상과 같은 응용서비스들이 증가하고 있다. 그러나, IPv4의 Best-Effort 서비스만으로는 이러한 요구사항을 충족시킬 수 없기 때문에 IETF에서는 QoS 보장을 위하여 DiffServ (Differentiated Services)와 Layer 2의 switching 기술을 사용함으로써 보다 빠른 패킷 전송 속도를 보이는 MPLS(Multi Protocol Label Switching)를 제시하였다. 또한 MPLS 망에 DiffServ의 기능을 추가시킨 DiffServ over MPLS 망이 제시되어 현재 논의 중에 있다. 본 논문에서는 Linux 환경에서 DiffServ over MPLS 망을 구축하여 IEEE 1394 응용서비스를 시험해 보고, 이를 통해 DiffServ over MPLS 시험망과 일반 인터넷 망의 성능을 비교 분석 해보고자 한다. 향후 본 논문을 이용하여 DiffServ over MPLS 망에서 활용될 수 있는 새로운 서비스를 개발하고 시험해 볼 수 있을 것이라고 생각한다.

1. 서 론

현재의 IP 프로토콜은 Best-Effort 방식으로 패킷을 전송하고 있다. 이는 네트워크 상의 모든 패킷들을 동일하게 취급하는 방식으로, 사용자의 등급이나 패킷의 중요성, 응용프로그램의 종류에 따른 QoS(Quality of Service)를 전혀 지원하지 않고 있다. 이러한 다양한 QoS를 제공하기 위해 IETF(the Internet Engineering Task Force)에서는 1994년에 IntServ(Integrated Service) 모델을 제안하였다[1]. IntServ WG(Working Group)은 voice/video와 같은 실시간 응용서비스들이 요구하는 QoS를 만족시킬 수 있는 라우터와 망 구조를 정의하였고 더 나아가서는 end-to-end에서의 QoS를 만족시킬 수 있는 기술을 제안하였으나, 인터넷 백본망의 모든 패킷 플로우에 대해 자원 예약시 필요한 정보 및 상태를 플로우마다 소프트 링크 상태로 유지하기 위해서는 라우터가 너무나 큰 메모리 공간과 계산 능력을 가져야 하기 때문에 실제 인터넷 망에서의 적용은 불가능

했다. 이러한 문제를 해결하기 위하여 IETF에서는 DiffServ WG을 만들어 각 트래픽 플로우에 대한 QoS 지원이 아니라 특정 패킷들의 집합에 대한 QoS를 보장해주는 방법을 제시하였다.

DiffServ는 기본적으로 DS 도메인의 입구 라우터에서 QoS를 만족시키기 위한 보다 복잡한 처리 과정을 수행하고 내부 라우터는 단순히 패킷에 정해진 처리 행동에 따라서 빨리 처리해주는 방법이다.

인터넷이 직면해 있는 또 다른 문제는 사용자 수가 급증하고 응용프로그램이 발전하면서 트래픽 양의 증가가 하드웨어의 발전 속도를 능가하고 있다. 때문에 한정된 자원인 대역폭을 보다 효율적으로 사용할 수 있는 방법이 필요하게 되었다. 이러한 인터넷 트래픽 문제를 해결하기 위해서 IETF에서는 Layer 3의 확장성을 그대로 적용하면서 Layer 2의 고속 패킷 교환 기능을 사용하는 MPLS(Multi Protocol Label Switching) 기술을 개발하였다. MPLS 망의 경계 라우터에서 고정 길이의 짧은 레이블

을 할당하고 중간 라우터는 단지 레이블을 교환하면서 출구 라우터까지 패킷을 전달할 수 있는 기능이다. 이 방식을 사용하면 중간 라우터에서는 IP계층까지 패킷이 올라가지 않기 때문에 고속으로 패킷을 전송할 수 있다.

DiffServ는 IP계층에서 패킷들을 분류하고 MPLS는 Data Link 계층에서 패킷들을 전송하기 때문에 두 기술을 쉽게 결합시켜 DiffServ over MPLS라는 새로운 서비스를 제공할 수 있다. 이렇게 DiffServ의 패킷 분류 기능과 MPLS의 패킷 전송 기능을 동시에 사용한다면 인터넷이 반드시 해결해야만 하는 중요한 두 문제인 응용서비스들의 QoS 보장에 대한 요구와 인터넷 트래픽 증가의 문제를 모두 해결할 수 있을 것이다.

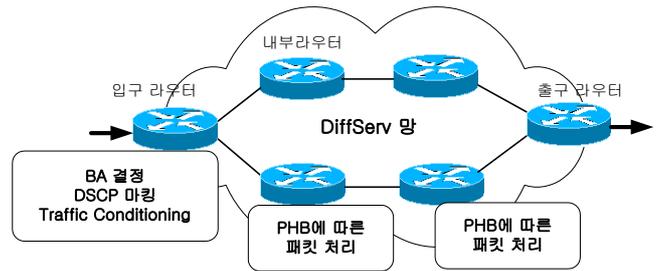
본 논문에서는 Linux 기반으로 MPLS 망에서 DiffServ 기능을 지원하는 DiffServ over MPLS(이하 DSMPLS) 시험망을 구축하고 IEEE 1394를 이용하여 디지털 비디오를 전송하는 실험을 해 보고자 한다.

2장에서는 관련 연구에 대해 살펴보고 3장에서는 MPLS 망에 DiffServ 기능을 추가하기 위해 필요한 사항들을 설명하였다. 4장에서는 Linux 기반의 PC 라우터를 이용하여 DSMPLS 시험망을 구축하였고 5장에서는 IEEE 1394를 이용한 디지털 비디오 데이터 전송을 통하여 시험망의 성능을 측정하였다.

2. 관련 연구

2.1 DiffServ

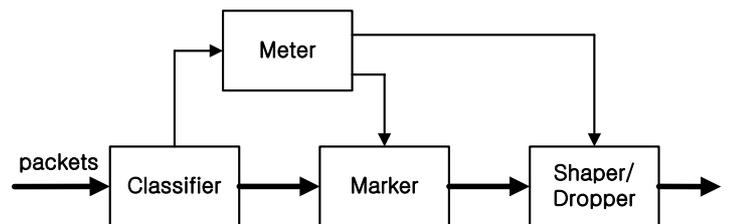
DiffServ망의 입구 라우터에서는 패킷 헤더의 특정 정보를 보고 차등화된 서비스를 위한 패킷들의 모임인 BA(Behavior Aggregate)를 결정한다. 각 BA에 따라서 망 내부에서 PHB(Per-Hop Behavior)를 결정하는 DSCP(DiffServ Code Point) 값을 패킷에 Marking하고 트래픽 조절 기능을 수행한다. 내부 라우터에서는 단순히 자원 할당과 PHB에 따른 패킷 처리를 수행하게 된다 [2]. 아래 <그림 1>은 DiffServ의 일반적인 구조를 보여주고 있다. 이렇게 DiffServ는 입구 라우터에서만 복잡한 기능을 수행하기 때문에 내부 라우터의 패킷 처리에 대한 부담을 줄여 처리 속도를 향상시켰다. DSCP는 IPv4의 헤더 부분의 Type of Service(ToS) 필드를 재정의하여 사용하고 있다[3].



<그림 1. DiffServ 망 구조>

DSCP 값과 PHB 매핑 문제는 DS 망 운영의 지역정책에 따라 달라 질 수가 있는데 IETF에서 권장하는 PHB의 종류에는 Default PHB, EF PHB, 그리고 AF PHB가 있다. Default PHB는 non DiffServ 패킷에 대해 기본적으로 서비스 해주는 PHB로써 현재의 Best-Effort 서비스와 같은 정도의 서비스이며 DSCP는 000000이다. EF PHB는 적은 손실, 지연, 지터와 대역폭 보장 서비스를 제공할 수 있는 PHB로써 양질의 서비스를 요구하는 트래픽을 서비스하기에 가장 적절하며 DSCP는 101110 (0x2e)이다. AF PHB는 망의 혼잡 및 폭주 상황에서도 트래픽의 최소한의 전송 속도를 보장하는 PHB이다. 이러한 AF PHB는 크게 4개의 서비스 클래스로 나누어지고 각 클래스에는 다시 3개의 다른 패킷 폐기 우선순위를 적용하게 된다[4][5].

경계 라우터에서 TC(Traffic Conditioning) 기능은 구체적으로 Classifier, Meter, Marker, Shaper/Policer등으로 구성되어 있다. Classifier는 입력 패킷들을 BA로 구분해 주며, Marker는 결정된 PHB에 따라 패킷에 DSCP값을 마킹한다. Meter는 패킷이 미리 정해진 트래픽 profile을 준수하는지 여부를 결정하고, 위배된 패킷에 대해서는 Shaper/Policer에서 조정 또는 폐기시키게 된다[2]. 아래 <그림 2>는 TC의 구조를 보여주고 있다.



<그림 2. DiffServ의 Traffic Conditioning 구조>

DiffServ 망 내의 각 라우터들은 PHB 별로 큐를 할당하고 버퍼링 함으로써 차등 서비스를

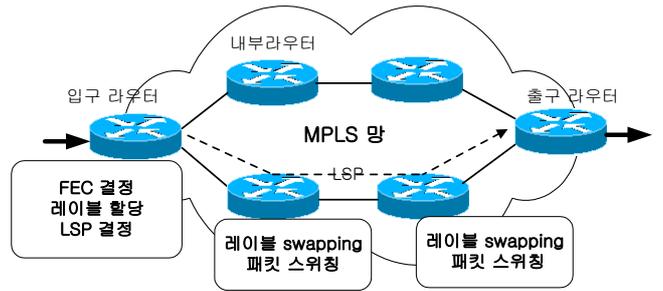
실행한다. EF PHB를 위해서는 CBWFQ(Class Based Weighted Fair Queuing), WRR(Weighted Round Robin), DRR(Deficit Round Robin) 등과 같은 큐잉 방법을 사용하고 AF PHB를 위해서는 각 AF 클래스에 따라 다른 패킷 폐기 방법을 적용할 수 있는 RED(Random Early Detection), WRED(Weighted RED) 등과 같은 여러 큐잉 메커니즘이 사용된다.

2.2 MPLS

MPLS는 패킷 전송을 고속화하기 위한 기술이며, 이를 위해 Layer 2의 교환 기술을 사용한다. 또한 망의 확장성을 제공하기 위해서 Layer 3의 라우팅 기술을 접목하였다. MPLS는 패킷을 전달하기 전에 반드시 입구 라우터에서 출구 라우터까지의 경로를 LSP(Label Switched Path)로 미리 설정하여야 하고, IP 주소 대신에 고정된 길이의 짧은 레이블을 사용하여 패킷을 전송한다. 즉 입구 라우터에서 패킷들을 FEC(Forwarding Equivalence Class)로 분류하고 이에 따라 레이블을 결정하게 되며, 그 레이블을 보고 미리 설정된 LSP를 선택하여 패킷을 전송하게 된다. 그러므로 패킷을 전송하기 위해 필요한 IP 헤더 처리 과정이 입구에서 한번만 수행되며 내부 라우터에서는 단지 레이블을 교환하면서 Layer 2의 고속전송을 하게 된다[6]. MPLS의 또 다른 특징 중에 하나는 기존의 인터넷 망에서는 가능하지 못했던 TE(Traffic Engineering)기능을 제공한다는 것이다. 이것은 단지 목적지 주소만을 보고 홉 단위로 라우팅을 수행하는 기존의 인터넷 프로토콜과는 달리 MPLS에서는 LSP를 설정할 때 네트워크 관리자가 선택적으로 Explicit Route를 설정할 수 있는 기능과 LSP 설정시 망 자원을 임의로 할당할 수 있는 기능을 제공함으로써 가능하다[7]. 또한 MPLS는 VPN(Virtual Private Network) 서비스를 제공하기에 적합하다. 기존 인터넷 망에서 VPN 서비스를 제공하기 위해서는 PPTP(Point-to-Point Tunneling Protocol), L2F(Layer 2 Forwarding), L2TP(Layer 2 Tunneling Protocol)과 같은 별도의 터널링 프로토콜을 제공해야 하지만, MPLS에서는 LSP에서 이용되는 레이블과 VPN 식별자 정보를 매핑시킴으로써 쉽게 VPN 서비스를 제공할 수 있다.

MPLS 라우터가 레이블을 교환하면서 패킷을 전달하기 위해서는 각 라우터가 사용하는 레이블 정보를 서로 주고 받아야하고 각 LSP가 차

등 서비스를 제공하기 위해서는 경로 설정 시에 필요한 QoS 정보들을 전달할 수 있게 하는 제어 프로토콜이 필요하다. 현재 IETF에서 표준화 단계에 있는 레이블 분배 프로토콜은 LDP(Label Distribution Protocol)와 기존의 RSVP를 확장한 RSVP-TE가 있다[8][9]. 아래 <그림 3>은 MPLS망의 일반적인 구조이다.



<그림 3. MPLS 망 구조>

2.3 IEEE 1394와 DV(Digital Video) 응용서비스

2.3.1 IEEE 1394의 개요

IEEE 1394는 고속의 시리얼 버스 인터페이스를 정의한 표준을 말한다. 이 기술은 1980년대 Apple 사의 "FireWire"에 근간을 두고 있는데, Apple은 자신들의 매킨토시 컴퓨터에 사용되는 SCSI 버스를 보충하는 저가격의 기술로 개발하였다. IEEE 1394의 가장 큰 장점으로는 빠른 전송 속도를 들 수 있으며 모드에 따라 100Mbps, 200Mbps, 400Mbps 등 세 가지의 속도를 낼 수 있기 때문에 고속의 대역폭을 요구하는 다양한 디지털 장비들 사이의 표준으로 주목되고 있으며, 따라서 많은 전자제품들이 IEEE 1394 인터페이스를 제공하기 시작하고 있다[10]. 몇 년 안에 캠코더, 비디오, 오디오, HDTV 등의 많은 장비들이 IEEE 1394 인터페이스를 제공하게 되며, 이들은 모두 아날로그가 아닌 고속의 디지털 데이터 전송에 기반으로 동작하게 된다. 또한 쌍방향 통신이 뛰어나기 때문에 PC를 이용한 화상 회의 등의 응용분야에도 적용이 용이하다.

2.3.2 디지털 비디오 응용서비스

디지털 비디오(DV)는 packet based video/audio 포맷이며 magnetic tape media에 영상과 음성을 저장하기에 최적화되어 설계되어 있고 작은 사

이즈의 tape media(6mm, 120min/cassette)와 8mm와 비교해서 적당한 가격으로 인해 가장 인기 있는 포맷이 되었다. DV 스트림 데이터는 인터넷 상에서 리얼타임 스트림을 위해 설계된 RTP(Realtime Transport Protocol)사용한다. DV 스트림 데이터는 3 byte ID 헤더를 포함한 80바이트 크기의 DIF block으로 구성되며 같은 RTP 패킷 안에 있는 DIF block들은 같은 비디오 프레임 안에 있는 것들이며 그 다음 비디오 프레임의 DIF block들과 같은 RTP 패킷 안에 존재할 수 없다. 그리고 DV 스트림은 RTP 타임 스탬프(time stamp)에 의해 한 프레임에서 다음 프레임으로 넘어가게 된다[10].

3. DSMPLS 개요

	입구 라우터	내부 라우터	출구 라우터
DiffServ	1. 패킷 검사 2. BA 결정 3. PHB 선택 4. DSCP 결정	DSCP를 참조하여 scheduling & dropping	-
MPLS	1. 패킷 검사 2. FEC 결정 3. LSP 결정 4. label 할당	label을 참조하여 swapping & switching	label 제거

<표 1. DiffServ와 MPLS의 비교>

DiffServ와 MPLS는 <표 1>과 같이 비교할 수 있다. 자세히 살펴보면, 입구 라우터에서 대부분의 작업을 수행하며 내부라우터에서는 상대적으로 단순한 작업만을 수행한다. 또한 수행계층이 DiffServ는 IP 계층이고 MPLS는 Data Link 계층이기 때문에 각각의 구조가 서로 충돌하거나 겹치는 부분이 없다. 이러한 이유로 두 서비스를 통합하는 것이 아주 용이하다. 기본적인 실행 시나리오를 살펴보면 입구 라우터의 IP 계층에서 DiffServ의 PHB를 선택하게 되고 Data Link 계층으로 내려보내게 된다. 여기서 PHB에 따라 MPLS의 레이블과 EXP 필드가 결정되어 해당 LSP에 따라 패킷 전송 서비스가 시작된다. 내부 라우터에서는 DiffServ의 관점으로 보면 MPLS 터널을 통과하게 되므로 특별히 DiffServ에선 처리해 줄 작업이 없다. 즉 DiffServ의 기능은 MPLS의 존재 여부에 관계없이 변함없다.

MPLS 망에서 DiffServ를 제공하기 위해서 해

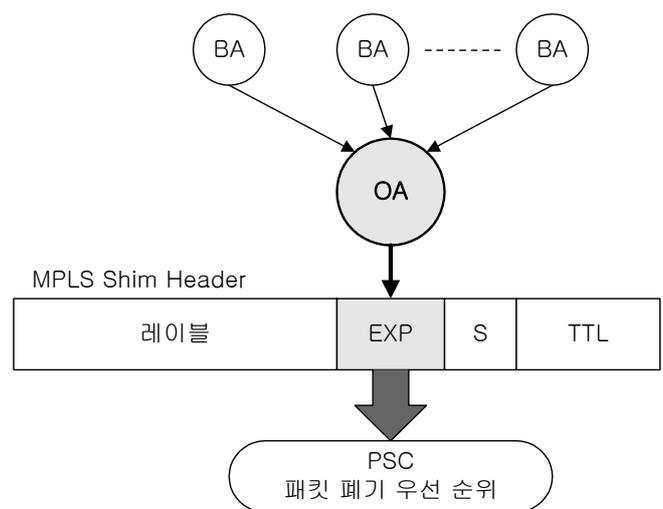
결해야 하는 문제들은 다음과 같은 것들이 있다[11].

- ▶ DiffServ에 정의되어 있는 PHB들을 MPLS LSP에 매핑시키는 방법
- ▶ DiffServ 패킷 전달을 위한 MPLS 라우터의 패킷 포워딩 엔진의 구조 변경
- ▶ DiffServ를 지원하는 LSP를 표현하기 위한 LSP 설정 방법

3.1 PHB의 LSP 매핑 방법

MPLS 망에서 DiffServ를 제공하기 위해서 제일 먼저 해결해야 하는 문제는 DiffServ의 BA들을 어떻게 MPLS LSP에 매핑 시키는 것이다. BA들을 LSP에 매핑하는 문제는 간단히 하나의 BA에 하나의 LSP를 설정하는 방법과 여러 BA들을 하나의 LSP에 매핑시키는 방법이 있을 수 있다. 이를 위해, IETF에서는 Label-Only-Inferred LSP(L-LSP)와 EXP-Inferred-PSC LSP(E-LSP)를 정의하고 있다. 두 LSP에 대해서 설명하기 앞서 DiffServ가 지원되는 MPLS에서 새로 추가된 용어들이 있는데 하나는 순서가 지켜져야 할 여러 BA들의 집합을 OA(Ordered Aggregate)로 정의하고 있고 OA에 속한 BA들에 적용할 하나 또는 그 이상의 PHB 집합을 PSC(PHB Scheduling Class)라고 정의하고 있다.

3.1.1 E-LSP

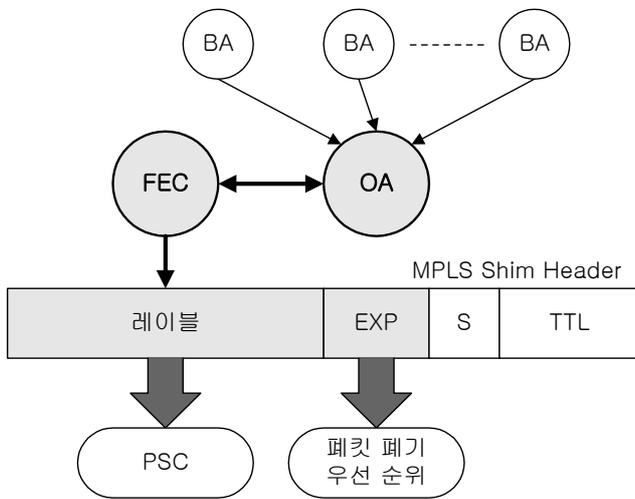


<그림 4. E-LSP의 BA 매핑 방법>

E-LSP는 여러 OA들에 대해서 하나의 LSP를 사용하는 것이다. FEC에 의해서 E-LSP는 결정

되고 하나의 E-LSP 내에서는 MPLS Shim 헤더의 EXP 필드를 이용하여 PHB를 결정한다. Shim 헤더의 EXP 필드가 3비트이므로 하나의 E-LSP에는 최대 8가지의 서로 다른 BA들을 전달할 수 있다. EXP 필드 하나로 PSC와 패킷 폐기 우선순위까지 결정되기 때문에 "EXP-Inferred-PSC LSP"라고 부른다. EXP 필드와 PHB의 매핑 방법은 E-LSP 설정 시에 명시해주는 방법과 미리 정의된 규칙에 따라 매핑하는 방법이 있다.

3.1.2 L-LSP



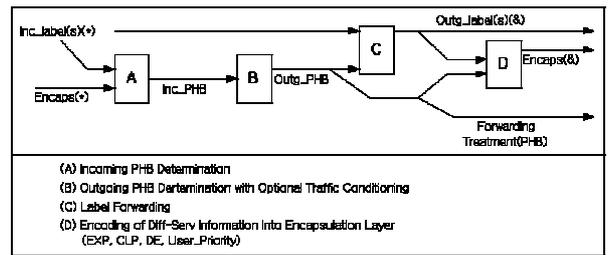
<그림 5. L-LSP의 BA 매핑 방법>

L-LSP는 <FEC, OA> 쌍에 대하여 하나의 LSP를 사용하는 방법이다. 따라서 LSP 설정 시에 하나의 PSC가 결정되고 각 LSR들은 레이블을 보고 해당 PSC를 매핑 시킬 수 있다. 그리고, 패킷 폐기 우선 순위는 MPLS Shim 헤더가 존재하면 EXP 필드를 보고 결정하며 Shim 헤더가 사용되지 않는 경우에는 해당 데이터 링크 계층에서 제공하는 패킷 폐기 우선 순위 필드를 참조하여 결정한다(예, CLP: MPLS-over-ATM, DE: over-Frame Relay, User_Priority: over-IEEE802.1). EXP를 고려하지 않고 단지 LSP의 레이블만으로 PSC가 결정되기 때문에 "Label-Only-Inferred-PSC LSP"라고 부른다.

3.2 MPLS 라우터의 패킷 포워딩 엔진의구조

하나의 LSP에 여러 OA들이 전송되기 때문에 DiffServ LSR의 레이블 교환 방식은 패킷의 BA에 따라 정해지며 DSCP 값을 직접 LSR이

얻을 수 없기 때문에 PHB를 MPLS 헤더에 인코딩하고 그 값에 따라 PHB를 결정하는 방식이 순수 DiffServ와 다르다. 따라서 DiffServ LSR 레이블 전달 구조는 변경되어야 한다. DiffServ를 지원하기 위해서 변경되어야 하는 MPLS LSR의 패킷 포워딩 구조는 먼저 PHB에 해당하는 패킷들을 구분할 수 있는 분류기(A), 출력 PHB를 결정할 수 있는 부분(B), Traffic Conditioning(TC), 레이블 교환 부분(C), 그리고 DiffServ 정보를 패킷에 설정하는 인코딩 기능(D) 등이 추가되어 져야한다. 아래 <그림 6>은 IETF에서 정의한 DiffServ가 지원되는 MPLS LSR의 레이블 포워딩 모델을 보여주고 있다. 그림에서 (*) 표시는 LSR이 MPLS ingress일 때, 입력 패킷이 레이블이 없을 수도 있다는 것을 의미하며, (&)는 MPLS egress LSR일 때, 출력 패킷이 레이블 없이 전달될 수 있다는 것을 의미하고 있다.



<그림 6. DSMPLS LSR의 패킷 포워딩 구조>

▶ 입력 PHB 결정(B)

MPLS LSR에 입력되는 패킷이 어느 PHB에 속하는지를 결정하는 패킷 분류기 역할을 수행한다. 구분 방법은 지원하는 DiffServ 터널링 모드에 따라 결정되는 레이블 스택 엔트리를 고려하는 방법과 IP 헤더의 DS 부분을 고려하여 결정하는 두가지 방법이 있다.

▶ TC 기능이 추가된 출력 PHB 결정(B)

출력 PHB를 결정하는 부분으로 가능하면 TC 기능도 수행한다. TC 기능은 부가적으로 수행되고 만약 TC 기능을 거치지 않으면 출력 PHB는 입력 PHB와 동일하다.

▶ 레이블 교환(C)

MPLS 패킷의 레이블 교환을 수행하는 부분으로 레이블이 부착되어 도착한 패킷은 ILM(Incoming Label Map)을 참조하여 하나 또는 여러 개의 NHLFE(Next Hop Label Forwarding

Entry)로 매핑될 수 있다. 레이블이 없는 패킷은 FTN(FEC-To-NHLFE) 맵을 참조하여 매핑한다. DiffServ를 지원하는 레이블을 나타내기 위한 문맥 정보를 다음과 같이 정의한다.

- '제공되는 PHB들'
- 'LSP 유형'(E-LSP, L-LSP)
- 입력 레이블을 위한 'Encaps→PHB 매핑'
- 출력 레이블을 위한 'Set of PHB→Encaps 매핑'

이러한 문맥 정보는 ILM과 FTN에 추가되고 LSP 설정 시에 구성되어진다. E-LSP 설정 시에 EXP ↔ PHB 매핑 정보를 명시하지 않은 E-LSP에 대해서는 미리 정의된 기본 PHB가 '제공되는 PHB들'에 설정되고 명시적으로 알려준 경우에는 전달받은 매핑 정보가 구성된다. L-LSP의 '제공되는 PHB들'은 LSP 설정 시에 반드시 명시적으로 전달받은 정보가 설정되어진다.

▶ DiffServ 정보의 인코딩(D)

DiffServ 관련된 정보를 해당하는 패킷에 설정하고 필요에 따라 적절히 변환해 주는 부분이다. 즉, MPLS Shim 헤더의 EXP, ATM CLP, FRDE, 802.1의 User_Priority 필드에 정보를 인코딩한다.

3.3 DiffServ 지원 LSP 설정 방법

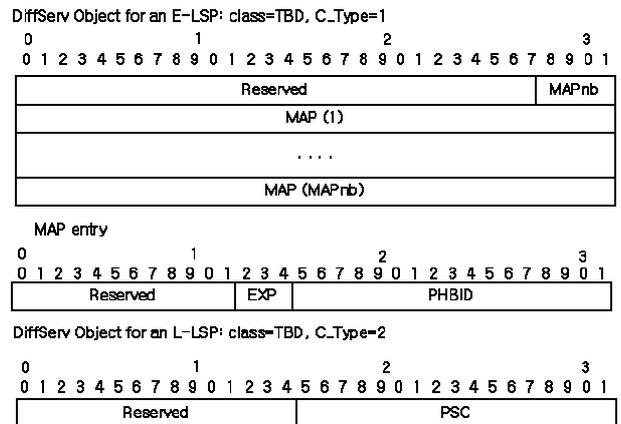
DiffServ를 지원하는 LSP 설정 방법은 기존의 제어 시그널링(LDP, RSVP-TE)을 약간 확장함으로써 가능하다. 즉, LDP(Label Distribution Protocol)은 DiffServ TLV를 정의하여 기존의 Label Request 메시지에 DiffServ TLV를 추가하여 전달함으로써 DiffServ LSP가 설정될 수 있으며, RSVP-TE는 DIFFSERV 객체를 새로 정의하여 PATH 메시지에 실어서 보냄으로써 가능하다.

3.3.1 DSMPLS를 위한 RSVP 확장

차등화 서비스 지원이 가능한 LSR이 E-LSP를 통해 차등화 서비스를 지원할 경우 DIFFSERV 객체에 대한 지원은 선택적이고, 만약 L-LSP를 통해 차등화 서비스를 지원할 경우, DIFFSERV 객체의 지원은 필수적이다. 다음 <그림 7>은 RSVP의 DIFFSERV 객체에 대한

정의이다. E-LSP의 class는 TBD이고, C_Type은 1이며 PHBID는 IANA에 정의된 값이다. L-LSP의 C_Type 값은 2이며 MAP 엔트리에 EXP 값이 없다. RSVP를 이용하여 E-LSP나 L-LSP를 설정할 때에는 기본적으로 세션 유형을 LSP_Tunnel_IpV4의 세션 유형과 LABEL_REQUEST 객체를 포함한 PATH 메시지를 보내어 설정한다. 이 PATH 메시지에 DIFFSERV 객체를 포함하여 레이블에 대한 DiffServ 문맥정보를 구성할 수 있도록 전달되어진다.

DIFFSERV 객체를 포함한 목적지 LSR은 PATH 메시지의 DIFFSERV 객체를 포함하지 않는 Resv 메시지를 보냄으로써 실제로 LSP가 설정되어진다. 레이블 요청이 받아들여지고 레이블이 할당되어지면 DiffServ LSR은 자신의 DiffServ 문맥 정보를 구성하고 패킷 처리 모듈의 매개변수 값을 요구된 차등 서비스를 제공할 수 있도록 버퍼 스케줄링 및 패킷 폐기 방법을 설정한다.



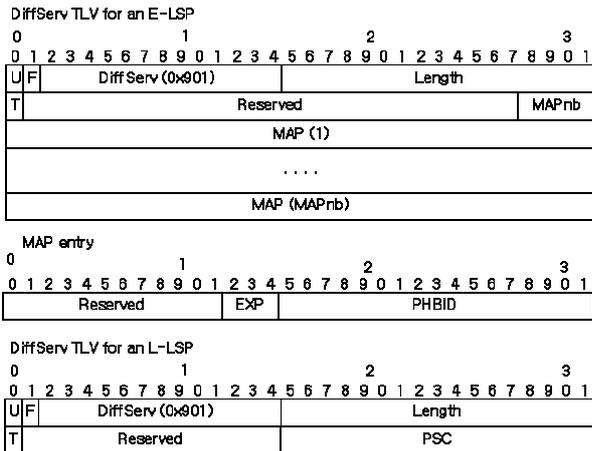
<그림 7. DIFFSERV 객체>

3.3.2 DSMPLS를 위한 LDP 확장

DiffServ를 지원하기 위해서는 LDP를 확장해야 한다. 이를 위해서 추가적인 DiffServ TLV에 대해서 IETF는 정의하고 있다. 이 DiffServ TLV는 미리 구성된 'EXP↔PHB 매핑'을 사용하는 E-LSP를 설정할 경우에는 생략되고, 명시적으로 알려주는 'EXP↔PHB 매핑'을 사용하는 E-LSP나 L-LSP를 설정할 때 Label 요청 메시지에 추가되어 전달되어진다.

아래 <그림 8>은 DiffServ TLV의 형식을 보여

주고 있다. E-LSP의 T는 0이며 MAPnb는 DiffServ TLV에 포함되는 MAP 엔트리의 수를 나타내며, 각 MAP 엔트리는 하나의 EXP 필드 값과 하나의 PHB를 매핑하는 것을 정의한다. L-LSP의 T는 1이며 단지 LSP에 의해 지원되는 PSC 값만을 가지고 있다.



<그림 8. DiffServ TLV>

4. DSMPLS 시험망 구축

DSMPLS 시험망 구축을 Linux를 이용하여 LSR을 3대 만들어 각각 입구 라우터, 내부 라우터, 출구 라우터로 설정하였다. 그리고, 완성된 시험망의 성능 측정을 위하여 IEEE 1394를 이용한 DV 전송서비스를 하였는데, 이를 위하여 경계 라우터에 필요한 장비들을 설치하였다.

4.1 Linux LSR 설치

시험망 구축을 위해 사용한 하드웨어 장비는 펜티엄 4 PC 3대에 100Mbps 이더넷 카드를 사용하였다. DSMPLS를 지원하는 PC 라우터 구축을 위하여 필요한 소프트웨어들을 살펴보면 아래와 같다[12].

- ▶ RedHat 계열의 Linux에 커널 버전은 2.4.9를 사용하였다.
- ▶ DiffServ의 기능은 커널 2.4 버전 이후 기본적으로 커널 내에 포함되어 있으므로, 특별히 설치할 필요는 없다.
- ▶ MPLS 기능은 Linux용 오픈 소스 개발 사이트인 SourceForge[13]에서 제공하는 MPLS 패치를 DiffServ와 연동을 위해 PHB를 레이블과 EXP로 변환시키는 기능을 추가시켜서 사용하

였다.

- ▶ 시그널링 프로토콜은 Alexey Kuznetsov가 개발한 RSVP 데몬에 E-LSP의 설정이 지원되도록 수정하여 사용하였다.
- ▶ 들어오는 패킷들을 분류하고 DSCP를 마킹하기 위해서 iptables 패키지를 설치하고 IPv4의 ToS 필드 부분에 DSCP를 마킹할 수 있도록 패치시켰다.
- ▶ LSP에 망 자원을 차등 분배하기 위해서 각각의 LSP에 스케줄링 큐를 연결하고 이를 통해 라우팅 할 수 있는 iproute2 패키지를 설치하였다. 스케줄링 큐의 설정은 iproute2 패키지 내의 'tc'를 사용하였다.
- ▶ 라우팅 기능은 특별한 라우팅 프로토콜을 사용하지 않고 Linux 자체의 기능을 사용하였다. 수신한 패킷을 다른 네트워크로 전송하기 위해서는 /proc/sys/net/ipv4/ip_forward 라는 파일의 내용을 '1'로 써주면 가능하다.

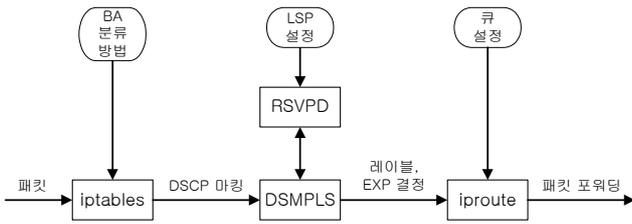
설치 순서는 커널에 MPLS 패치와 iptables의 DSCP 마킹을 위한 패치를 먼저 실행해야 한다. 그리고, 커널에 필요한 기능들을 설정하고 컴파일 한 후 다른 소프트웨어들을 설치한다. 커널에 설정해야 할 기능들은 아래 <표 2>와 같다.

```

Code maturity level options
- Prompt for development and/or incomplete code/drivers [y]
Networking options
- Kernel/User netlink socket [y]
  - Routing messages [y]
- Network packet filtering (replaces ipchains) [y]
  - Network packet filtering debugging [y]
- TCP/IP networking [y]
  - IP: advanced router [y]
    - IP: policy routing [y]
    - IP: use netfilter MARK value as routing key
  - IP: NetFilter Configuration
    - Connection tracking (required for masq/NAT) [y]
    - IP tables support (required for filtering/masq/NAT) [y]
      - netfilter MARK match support [y]
      - TOS match support [y]
      - DSCP match support [y]
      - Packet filtering [y]
      - Full NAT [y]
        - MASQUERADE target support [y]
      - Packet mangling [y]
        - TOS target support [y]
        - Full TOS target support [y]
        - MARK target support [y]
- Multi Protocol Label Switching - MPLS [y]
  - MPLS Ingress Policing [y]
  - MPLS Statistics [y]
  - MPLS TC extensions [y]
- QoS and/or fair queueing
  - all options [y]
  
```

<표 2. Linux 커널 설정 옵션>

Linux LSR 내에서의 패킷의 흐름을 설치한 소프트웨어들의 연동 구조로 살펴보면 아래 <그림 9>와 같다.



<그림 9. Linux LSR의 구조>

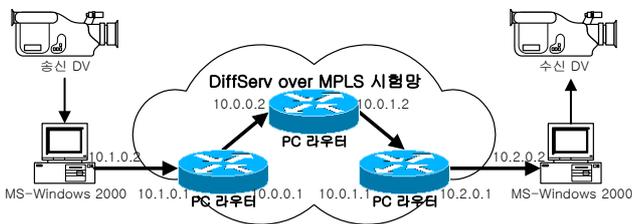
4.2 DV 전송 서비스를 위한 하드웨어 설치

IEEE 1394를 이용한 DV 전송 서비스의 시험에 사용한 하드웨어 장비의 목록이다. 아래의 장비들은 DV의 송신과 수신을 위해 각 2개씩 필요하다.

- 펜티엄 3 PC
- DV의 송수신에 사용할 Sony TRV900 디지털 비디오 카메라
- DV 카메라와 PC간의 IEEE 1394 인터페이스와 연결에 필요한 FireBird DV 1394 card (Texas Instruments chip: TSB12LV26) & cable (4-6 pin)
- 100Mbps 이더넷 카드
- 사운드 카드

위의 하드웨어를 장착한 각각의 PC에 MS-Windows 2000을 운영체제로 설치하고 DV의 전송을 위해 DVTS(Digital Video Transport System)을 설치하였다[14].

위의 장비들을 이용하여 구축한 시험망은 아래 <그림 10>과 같다.



<그림 10. DSMPLS 시험망>

5. IEEE 1394 DV의 전송 시험 및 결과

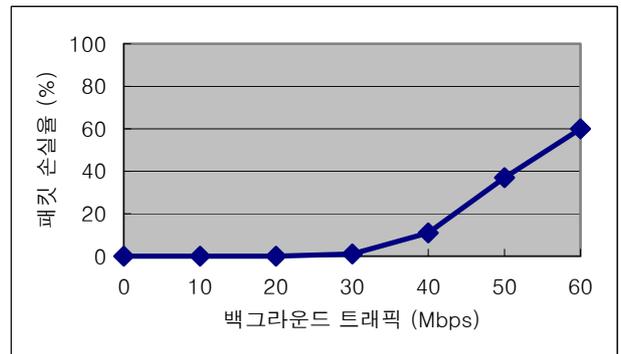
4절에서 구축한 DSMPLS 시험망 위에 DV 데이터 전송을 위해 EF PHB를 설정하고 다른 패킷들을 위해 Default PHB를 설정하여 패킷을 전송했을 때 DV 데이터의 손실율을 계산하여

시험망의 성능을 측정 후 일반 인터넷 망과 성능을 비교 분석하였다.

5.1 일반 인터넷 망에서의 DV 송수신

일반 인터넷 망의 성능을 측정하기 위한 시나리오는 아래와 같다.

1. 송신측 PC와 수신측 PC의 DVTS 응용프로그램을 실행시켜서 각각 sender와 receiver로 설정한 후 송신측 DV 카메라에서 수신측 DV 카메라로 테스트용 DV의 전송을 시작한다.
2. 시험망에 DV 이외의 다른 트래픽이 없기 때문에 입구 라우터에서 출구 라우터까지 백그라운드 트래픽을 발생시킨다.
3. 백그라운드 트래픽의 양을 10Mbps 단위로 점차 증가시킨다.
4. 위의 상황에서 발생하는 DV 패킷의 손실율을 측정 기록한다.



<그림 11. 일반 인터넷 망에서의 패킷 손실율>

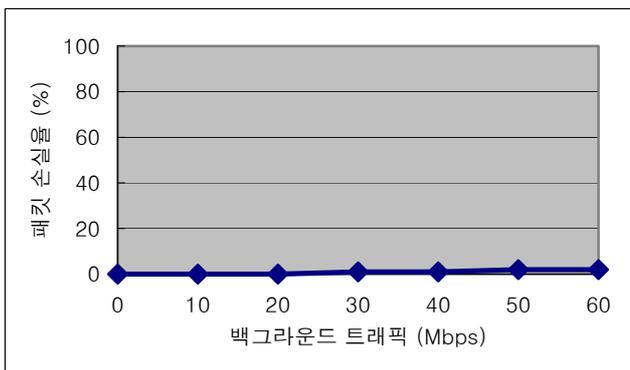
위의 <그림 11>은 시나리오에 따른 일반 인터넷 망에서의 백그라운드 트래픽 증가에 따른 패킷 손실율을 나타내고 있다. 이처럼 QoS 보장이 없는 망의 경우 망에 흐르는 트래픽의 양이 적을 때는 패킷의 손실이 발생하지 않지만, 양이 점차 증가하여 망의 성능을 능가하는 어느 순간부터 패킷 손실율은 기하 급수적으로 증가함을 알 수 있다.

5.2 DSMPLS 시험망에서의 DV 송수신

DSMPLS 시험망의 성능을 측정하기 위해 기본적으로 설정한 사항은 아래와 같다.

1. 입구 라우터에서 출구 라우터까지 LSP를 설정한다.
2. DSCP가 EF PHB로 마킹되어 오는 패킷에 대해 EXP 필드를 '1'로 마킹하도록 설정한다. 즉, PHB \leftrightarrow EXP를 'EF PHB (DSCP : 0x2e) \leftrightarrow 1'로 설정한다.
3. EF PHB와 Default PHB를 위한 스케줄링 큐를 설정한다. EF PHB를 위한 큐는 TBF(Token Bucket Flow) 큐를 사용하였고 Default PHB를 위해서는 RED(Random Early Detection) 큐를 사용하였다.
4. 송신측 PC에서 수신측 PC로 가는 DV 패킷의 DSCP를 EF PHB 값인 0x2e로 마킹하도록 설정하고 백그라운드 트래픽은 Default PHB 값인 0x00로 마킹하도록 설정한다.
5. DSCP 0x2e로 마킹한 패킷들을 하나의 FEC로 묶어서 1에서 설정한 LSP를 통하여 전송되도록 한다.
6. 송신측 DV 카메라에서 수신측 DV 카메라로 테스트용 DV의 전송을 시작한다.

이렇게 설정되어 있는 상황에서 EF PHB와 Default PHB의 우선순위 차이를 보기 위하여 EF PHB의 트래픽 허용량을 35Mbps로 보장해 주면서 백그라운드 트래픽의 양을 10Mbps 단위로 점차적으로 증가시키면서 DV 패킷의 손실율을 살펴 보았다.

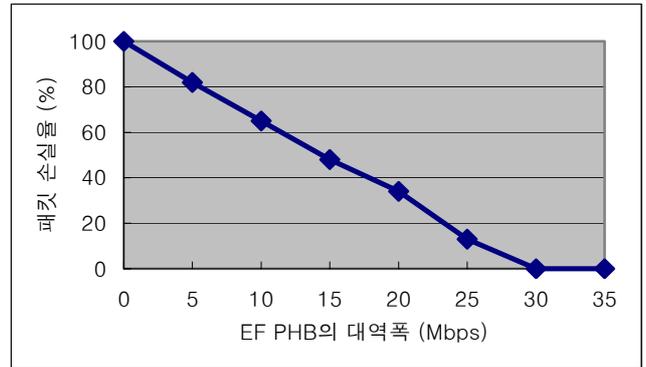


<그림 12. DSMPPLS 망에서 백그라운드 트래픽에 따른 EF PHB의 패킷 손실율>

위의 <그림 12>에서 알 수 있듯이 백그라운드 트래픽의 양에 관계없이 EF PHB의 트래픽은 항상 보장됨을 알 수 있다.

DSMPPLS 시험망에서 두 번째 실험은 입구 라우터에서 출구 라우터까지 백그라운드 트래픽의

양을 60Mbps로 설정한 후 EF PHB를 위해 사용한 TBF 큐의 대역폭을 5Mbps 단위로 증가시켜 패킷 손실율을 살펴 보았다. 아래 <그림 13>은 DSMPPLS 시험망에서 DV를 위한 EF PHB의 대역폭을 점차적으로 증가시켰을 경우의 패킷 손실율을 나타내고 있다.



<그림 13. DSMPPLS 망에서 EF PHB의 대역폭 변화에 따른 패킷 손실율>

EF PHB를 위해 설정한 대역폭은 다른 PHB에서 사용할 수 없다. 그러므로 Default PHB로 설정된 백그라운드 트래픽의 증감 여부에 상관없이 EF PHB의 대역폭은 보장되며 패킷 손실율은 거의 변동이 없다. 위의 그림을 살펴보면 DVTS의 요구 대역폭은 30Mbps이며 이를 보장해 주었을 때는 망내의 트래픽이 매우 혼잡한 상황에서도 원활한 패킷 전송이 가능함을 보여 주고 있다. 즉, 위의 실험 결과에서 알 수 있듯이 일반 인터넷 망에서 보장해주지 못하는 QoS를 고대역폭의 전송속도를 요구하는 응용서비스에게 보장해 줌으로써 보다 좋은 결과를 얻을 수 있음을 알 수 있다. 그러나, 각각의 PHB에 적절한 대역폭이 보장되지 않을 경우 기존의 인터넷 망 보다 못한 결과를 보일 수도 있으므로 망의 특성에 따라 적절한 대역폭의 분할이 요구된다.

6. 결론 및 향후 연구과제

본 논문에서는 Linux 기반의 PC를 이용하여 DSMPPLS 시험망을 구축하고 고대역폭의 전송속도를 요구하는 IEEE 1394를 이용한 DV 전송 서비스를 QoS를 보장하여 수행해 보았다. DSMPPLS 망을 이용해야 할 필요성이 있는 응용서비스를 개발할 경우 본 논문에서 구축한

시험망을 이용하여 테스트 할 수 있기 때문에 보다 빠르고 효율적인 개발 및 비용절감 효과를 볼 수 있을 것이라고 생각한다.

본 논문에서 구축한 시험망에 사용한 시그널링 프로토콜은 RSVP-TE이며 DiffServ를 지원하는 LSP로는 E-LSP를 사용하였다. 향후에는 CR-LDP를 시그널링 프로토콜로 사용하고 L-LSP를 지원할 수 있도록 수정하여 보다 완벽한 DSMPLS PC 라우터를 만들 계획이다. 또한 DSMPLS를 지원하는 PC 라우터에 사용된 각각의 제어프로그램들과 운영자 사이에 통합설정 프로그램을 구현함으로써 DSMPLS 시험망의 운영을 보다 일괄적이고 쉽게 설정할 수 있도록 하여 사용자의 편의를 도모하고자 한다.

참고문헌

- [1] Braden, R., Clark, D. and S. Shenker, "Integrated Services in the Internet Architecture: an Overview", RFC 1633, June 1994.
- [2] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z. and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, December 1998.
- [3] Nichols, K., Blake, S., Baker, F. and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.
- [4] Heinanen, J., Baker, F., Weiss, W. and J. Wroclawski, "Assured Forwarding PHB Group", RFC 2597, June 1999.
- [5] Jacobson, V., Nichols, K. and K. Poduri, "An Expedited Forwarding PHB", RFC 2598, June 1999.
- [6] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.
- [7] Awduche, D. et al., "Requirements for Traffic Engineering over MPLS", RFC 2702, September 1999.
- [8] Andersson et al., "LDP Specification", RFC 3036.
- [9] Awduche, D., Berger, L., Gan, D., Li, T., Swallow, G. and V. Srinivasan, "RSVP-TE: Extensions to RSVP for LSP Tunnels," RFC 3209, December 2001.
- [10] A. Ogawa, K.Kobayashi, et al., "Design and Implementation of DV based video over RTP", PVW2000, 2000
- [11] "MPLS Support of Differentiated Services",

- draft-ietf-mpls-diff-ext-09, September 2001.
- [12] Pim Van Heuven, "RSVP-TE daemon for DiffServ over MPLS under Linux", <http://dsmpls.atlantis.rug.ac.be>
 - [13] "Source Forge", <http://www.sourceforge.net>
 - [14] "DV Stream on IEEE 1394 Encapsulated into IP", <http://www.sfc.wide.ad.jp/DVTS>
 - [15] Srinivas Vegesna, "IP Quality of Service", Cisco Press, 2001
 - [16] Kalevi Kilkki, "Differentiated Services for the Internet", MTP, 1999



권 호 한
1999.2 전남대학교 전산학 학사
2001 - 현재 전남대학교 전산학 석사과정
관심분야 : MPLS, DiffServ



윤 상 식
1996 전남대학교 전산학 학사
1999 전남대학교 전산통계학 석사
1999 - 현재 전남대학교 전산학 박사과정
관심분야 : MPLS, TE, QoS



최 덕 재
1982 서울대학교 컴퓨터 공학 학사
1984 한국과학기술원 전산학 석사
1995 Univ.of Missouri-Kansas 컴퓨터 망 박사

1984-1996 전남대학교 전자계산소 운영부장
1997-1998 전자통신연구원 통신시스템 연구단 위탁 연구원
1996-2000 전남대학교 전산학과 부교수
2001.1 - 현재 전남대학교 전산학과 교수
관심분야 : 네트워크 관리, 서비스 관리, TMN, 차세대 인터넷