

MPLS 망 관리와 GSMP 연동 모델

윤창종 김진년 권태현 김춘희 차영욱
 안동대학교 컴퓨터공학과
 shion@comeng.andong.ac.kr

Interworking Model for MPLS Network Management and GSMP

Chang-Jong Yoon Jin-Nyun Kim Tae-Hyun Kwon Choon-Hee Kim Young-Wook Cha
 Dept. of Computer Engineering, Andong National University

요 약

MPLS는 계층 3과 계층 2 기술의 결합으로 고속의 패킷 전송 및 다양한 트래픽 엔지니어링을 가능하게 한다. GSMP는 마스터-슬레이브 관계를 형성하는 IETF의 개방형 인터페이스이다. 본 논문에서는 MPLS 네트워크의 망 관리와 GSMP의 연동 필요성 및 모델을 제안하며 제안된 연동 모델의 실현성을 확인하기 위하여 구현한 컨트롤러에 대하여 기술한다.

1. 서론

GSMP(General Switch Management Protocol)[1]는 레이블 스위치와 컨트롤러 사이에서 연결, 구성, 장애, 성능관리 및 동기화 기능을 제공할 수 있는 개방형 인터페이스이다. GSMP가 적용된 개방형 인터페이스에서 망 관리를 위한 SNMP(Simple Network Management Protocol) 에이전트가 GSMP 마스터와 슬레이브의 어느 위치에서 동작하는지는 명확하게 규정되어 있지 않다. 본 논문에서는 MPLS(Multi-Protocol Label Switching) 네트워크에서 LSR(Label Switch Router)의 레이블 스위치 및 컨트롤러 사이에 GSMP가 사용되는 경우에 망 관리와 GSMP의 연동 필요성 및 모델을 제안한다.

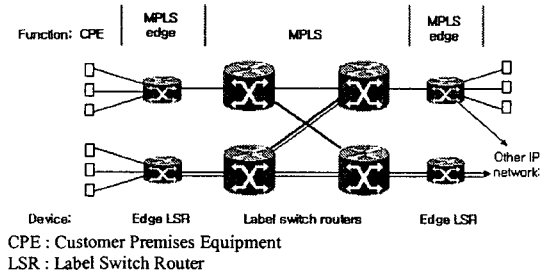
본 논문의 2장에서는 MPLS 기술과 MPLS의 망 관리에 대하여 소개하고, 3장에서는 개방형 인터페이스와 GSMP의 표준화 동향을 기술한다. 4장에서는 MPLS 망 관리와 GSMP의 연동 모델 및 시나리오를 제안하며, 본 논문에서 제안된 연동 모델의 실현성을 확인하기 위하여 구현한 LSR의 컨트롤러에 대하여 기술한다. 마지막으로 5장에서는 본 논문의 결론 및 향후 연구 과제에 대하여 기술한다.

2. MPLS

MPLS 기술은 계층 3의 라우팅과 계층 2의 레이블 스위칭 기술의 결합이라고 볼 수 있다. 레이블 스위핑에 기반한 고속의 포워딩 기술을 결합함으로써, 고속 패킷 전송이 가능해지고, 다양한 서비스들을 레이블에 매핑함으로써 QoS 보장 서비스를 제공할 수 있게 된다. 따라서 빠른 속도로 증가하고 있는 인터넷 사용자 수와 다양한 서비스의 요구에 적절히 대응할 수 있는 MPLS 기술이 고속의

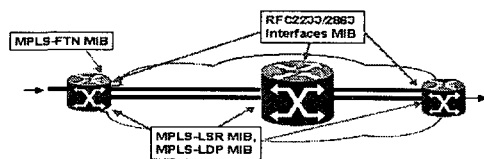
인터넷 백본 망을 구성하기에 적합한 방안으로 대두되고 있다[2].

MPLS 네트워크는 그림 1과 같이 코어 LSR이 가운데 위치하고 주위에 에지 LSR이 위치하게 된다. MPLS 네트워크에 접속된 CPE(Customer Premises Equipment)들은 MPLS보다는 기존의 IP 포워딩이 구동되는 라우터나 LAN 스위치이다[3][4].



<그림 1> MPLS 네트워크 구조

MPLS 망 관리를 위하여 정의되고 있는 MIB (Management Information Base)에는 그림 2 와 같이 MPLS-FTN(FEC To Next hop forwarding label entry) MIB[5], MPLS-LSR MIB[6] 및 MPLS-LDP MIB 가 있다. MPLS-FTN MIB 는 입구(ingress)의 에지 LSR 에서 적용되며, MPLS-LSR MIB 는 MPLS 망의 모든 LSR 에 적용된다.



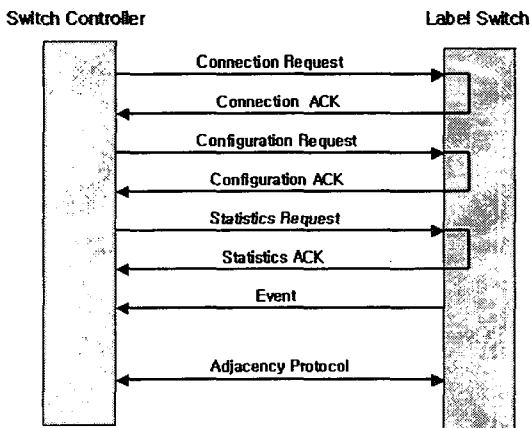
<그림 2> MPLS 네트워크의 MIB
 MPLS 입구의 에지 LER은 FEC(Forwarding Equiva-

lence Class)에 속한 패킷들을 FTN 매핑을 통하여 해당 LSP로 연관시키게 된다. MPLS-LDP MIB는 레이블 할당 및 분배를 위한 LDP 신호 프로토콜을 위하여 정의 되었다. MPLS LSR MIB는 LSP의 설정 및 관리를 위하여 인터페이스, 세그먼트 및 크로스 컨넥트 관련 정보를 정의하고 있다.

3. 개방형 인터페이스와 GSMP

통신 분야의 개방형 인터페이스는 1996년 콜롬비아 대학에서 개방형 네트워크에서의 제어 이슈들에 관한 연구를 활성화하기 위하여 OpenSig 워크샵이 개최되었다. IEEE P1520, Parlay Group, Multiservice Switching Forum, International Softswitch Consortium 등에서 기능 그룹과 그룹 사이 또는 평면들 사이의 표준 API를 각 그룹의 목적에 따라서 정의하고 있다. IETF에서는 개방형 인터페이스를 위하여 많은 프로토콜을 정의하고 있는데 대표적인 것으로는 MEGACO(Media Gateway Control), COPS(Common Open Policy Service), GSMP 프로토콜 등이 있다[7].

IETF의 GSMP는 그림 3과 같이 레이블 스위치와 컨트롤러 사이에서 연결, 구성, 장애, 성능관리 및 동기화 기능을 제공하는 개방형 인터페이스이다. GSMP 프로토콜 버전의 합의, 상태 동기화, GSMP 메시지가 교환될 링크 상의 송수신 포트 및 인스턴스 정보를 교환하는 Adjacency 프로토콜은 컨트롤러나 스위치 어느 곳에서든 먼저 활성화 될 수 있다. 반면에 연결, 구성 및 성능 관리를 위한 메시지들은 컨트롤러에 의해 생성되며, 스위치는 컨트롤러가 보낸 메시지에 대한 응답을 수행한다. 장애 관리를 위한 이벤트 메시지는 스위치에 의해 생성되며, 스위치 컨트롤러는 응답 메시지를 보내지 않는다[1].



<그림 3> GSMP 기능

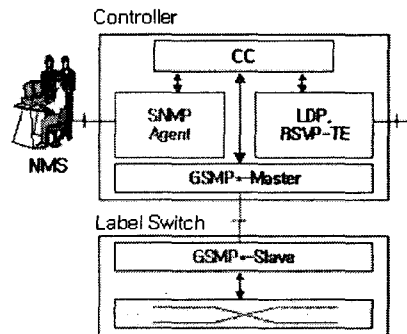
4. MPLS 망 관리와 GSMP 연동

4.1 연동 모델

GSMP 컨트롤러는 레이블 스위치의 모든 자원 상태를 알고 있다는 전제 하에서 GSMP 프로토콜이 설계되었고 또한 동작한다. GSMP가 적용된 개방형 인터페이스에서 망 관리를 위한 SNMP 에이전트의 위치가 GSMP 마스터와 슬레이브의 어느 위치에서 동작하는 지는 명확하게 규정되어 있지 않다.

SNMP 에이전트가 레이블 스위치에 위치하여 NMS를 통한 연결을 설정하는 경우에는 GSMP 컨트롤러가 레이블 스위치의 모든 자원 상태를 알고 있다는 전제가 허물어지게 된다. 이를 극복하기 위하여서는 망 관리를 통한 연결의 설정 및 해제 시에 레이블 스위치가 매번 이러한 상황을 GSMP 컨트롤러에게 통보하는 것이다. 다른 대안은 망 관리를 통하여 설정 및 해제되는 구성연결(provisioned connection)과 신호 프로토콜을 이용하여 설정 및 해제되는 동적연결을 위한 자원들을 분리하여 사용하는 방안이 있을 수 있다. 구성연결과 동적연결을 위한 자원의 분리는 자원의 이용면에서 비효율적인 방법이라고 판단된다[8].

GSMP의 개방형 인터페이스는 연결관리 뿐만 아니라, 장애관리, 성능관리 및 구성관리를 위한 절차들이 정의되어 있다. GSMP의 개방형 인터페이스에서는 이러한 관리 기능들의 지능을 컨트롤러에서 수행하게 함으로 레이블 스위치는 디바이스 드라이버 및 기본적인 시스템 관리 정도의 기능만을 요구하게 된다. 그러나 SNMP 에이전트가 레이블 스위치에 위치하게 되면 망 관리 메니저 및 GSMP 컨트롤러와의 인터페이스를 위한 관리 기능들을 가져야 한다. 이는 GSMP 개방형 인터페이스에서 추구하는 레이블 스위치의 기능을 단순화시키려는 의도에 위배되는 접근 방법이다.



NMS : Network Management System
CC : Connection Controller

<그림 4> MPLS 망 관리와 GSMP의 연동 모델

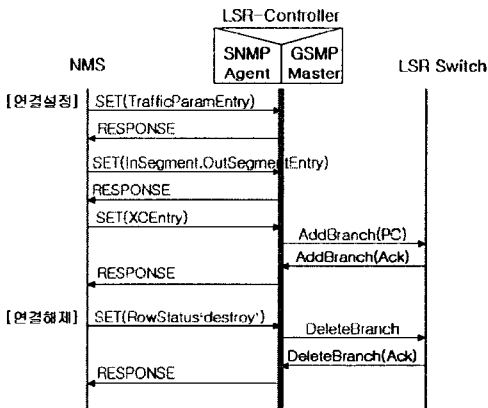
위와 같은 문제점들은 SNMP 에이전트가 그림 4와 같이 컨트롤러에 SNMP 에이전트와 GSMP의 마스터가 위치한다면 자연스럽게 해결될 수 있다. 즉 SNMP 에이전트는 NMS(Network Management Sys-

tem)로부터의 망 관리 요구를 수신하여 이를 GSMP를 통하여 레이블 스위치로 전달하는 연동 모델이다. 이를 위해서는 GSMP가 구성연결과 동적연결을 모두 관리할 수 있도록 확장되어야 한다. 또한 GSMP의 구성, 성능 및 장애관리 기능에 대한 연동이 가능하도록 MPLS MIB의 확장이 요구된다.

4.2 연동 시나리오

4.2.1 연결관리

그림 5는 망 관리를 통하여 LSR에 구성연결의 설정과 해제를 위한 SNMP와 GSMP의 연동 절차를 나타낸다. NMS가 트래픽 파라미터 및 세그먼트 엔트리를 설정한 후에 크로스 컨넥트 엔트리의 설정을 요구하면, 컨트롤러는 GSMP의 Add Branch 메시지를 이용하여 레이블 스위치에게 구성연결의 설정을 요구하게 된다. GSMP는 기본적으로 신호 프로토콜과 연동을 통한 동적연결을 설정 및 해제하므로 Add Branch 메시지에서 요구되는 연결 요청이 구성연결인지 동적연결인지를 나타내는 플래그가 요구된다.



PC : Provisioned Connection

<그림 5> SNMP와 GSMP의 연결관리 연동

연결 해제 과정은 NMS가 크로스 컨넥트 엔트리의 상태 칼럼에 destroy 값을 설정하여 연결의 해제를 요구하게 되며, 이를 수신한 컨트롤러는 GSMP의 Delete Branch 메시지를 이용하여 구성연결의 해제를 스위치에게 요구하게 된다.

4.2.2 구성관리

MPLS-LSR MIB에서는 인터페이스의 MPLS 지원 여부를 나타내기 위한 mplsInterfaceConfTable을 지원하고있다. GSMP에는 스위치, 특정 포트 및 모든 포트 그리고 서비스와 관련한 구성관리 기능을 제공한다.

컨트롤러에서 SNMP와 GSMP의 연동을 지원하기 위해서는 GSMP에서 제공되는 구성관리 메시지에 기반한 MIB의 확장이 요구된다. 다음은 GSMP의 Switch Configuration 메시지와와의 연동을 위하여 확장한 anuMplsSwitchConfTable로써 각 엔트리들은

스위치의 분할 식별자를 인덱스로 사용한다.

```

anuMplsSwitchConfTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF AnuMplsSwitchConfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table specifies switch configuration."
    ::= { anuMplsLsrObjects 1 }
    
```

```

anuMplsSwitchConfEntry OBJECT-TYPE
    SYNTAX      AnuMplsSwitchConfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This entry specifies the global (not port specific) configuration
        for the switch partition."
    INDEX       { anuMplsSwitchPartitionIndex }
    ::= { anuMplsSwitchConfTable 1 }
    
```

4.2.3 성능관리

MPLS-LSR의 MIB에는 인터페이스, 입력 세그먼트 및 출력 세그먼트들에 대한 성능 관리를 위한 세 개의 테이블을 정의하고 있다. GSMP에서는 Port Statistics 및 Connection Statistics 메시지를 이용하여 포트 및 연결에 관한 성능관리를 수행한다.

MPLS-LSR MIB의 mplsInterfacePerfTable의 각 엔트리들은 GSMP의 Port Statistic 메시지와 연동이 가능하다. GSMP의 Connection Statistics 메시지에 있는 입력 및 출력 카운터 정보 요소들은 MPLS-LSR MIB의 mplsInSegmentPerfTable 및 mplsOutSegmentPerfTable의 각 엔트리들과 연동이 되어야 한다.

4.2.4 장애관리

GSMP에는 장애관리를 위하여 Port Up, Port Down, Invalid Label, New Port, Dead Port 및 Adjacency Update와 같은 이벤트 메시지를 정의하고 있다. GSMP 컨트롤러는 이들 이벤트 메시지가 레이블 스위치로부터 통보되면 이를 SNMP Trap 메시지로 망 관리 매니저에게 통보하여야 한다.

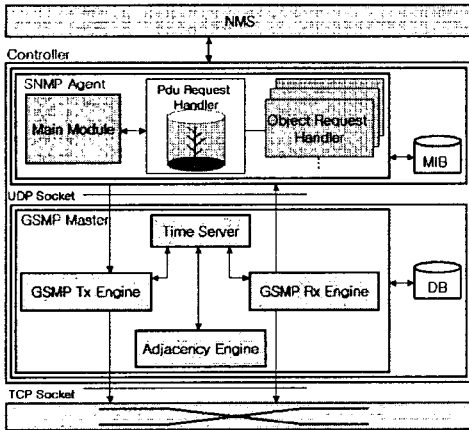
MPLS-LSR MIB에는 크로스 컨넥트 생성 및 해제와 관련된 NOTIFICATION만이 정의되어 있다. 표 1은 GSMP 이벤트 메시지와의 연동을 위하여 새로이 정의한 NOTIFICATION들이다.

<표 1> GSMP 이벤트와 NOTIFICATION 연동

GSMP 이벤트	NOTIFICATION
Port Up	AnuMplsPortUp {partition ID, port no, port session no}
Port Down	AnuMplsPortDown {partition ID, port no, port session no}
Invalid Label	AnuMplsInvalidLabel {partition ID, port no, port session no, label}
New Port	AnuMplsNewPort {partition ID, port, port session no}
Dead Port	AnuMplsDeadPort {partition ID, port, port session no}
Adjacency Update	AnuMplsAdjacencyUpdate {partition ID, code}

4.3. 망 관리와 GSMP의 연동 구현

본 절에서는 본 논문에서 제안된 연동 모델의 실현성을 확인하기 위하여 구현한 LSR의 컨트롤러에 대하여 기술한다. 연동 모델의 전체 구현 구조는 그림 6과 같다.



<그림 6> 연동 모델의 구현 구조

SNMP 에이전트는 AdventNet사의 Agent Toolkit을 사용하여 구축했으며, 메인 모듈, PDU Request Handler, ObjectRequestHandler 등으로 이루어진다. 메인 모듈은 NMS로부터 받은 명령의 타당성을 검사하고 오류가 없을 때 PDU Request Handler에게 명령을 보낸다. PDU Request Handler의 등록 트리에는 모든 ObjectRequestHandler(Listener)가 등록되어 있어서 명령을 받았을 때 해당 리스너를 찾아 명령을 처리한다. ObjectRequestHandler는 MIB의 테이블 정보들을 파일형태로 유지하며 PDU Request Handler의 등록 트리에 오브젝트 식별자로 등록되어 있다[9].

GSMP 마스터는 쓰레드로 구성된 4개의 엔진으로 이루어진다. SNMP 에이전트에서 전달되는 메시지를 받아서 스위치로 전송하는 GSMP Tx 엔진과 스위치로부터 응답 메시지를 수신하는 GSMP Rx 엔진이 있으며, Time Server와 Adjacency 엔진은 GSMP 컨트롤러와 스위치의 동기를 맞추고, 접속의 손실을 감지한다. GSMP 마스터는 스위치와의 상호작용에 의하여 생성 및 수정된 연결, 구성, 이벤트 정보 등을 JDBC(Java Data Base Connectivity)와 연계하여 데이터베이스에서 관리한다.

5. 결론

MPLS는 계층 3의 라우팅과 계층 2의 레이블 스위칭 기술의 결합을 통하여 고속의 패킷 포워딩 및 다양한 QoS 보장 서비스를 제공할 수 있다. MPLS의 LSR에서 전달요소과 제어요소 사이에 적용이 가능한 GSMP는 연결, 구성, 장애, 성능관리

및 동기화 기능을 제공하는 개방형 인터페이스이다.

본 논문에서는 MPLS 망 관리와 GSMP 연동 필요성 및 연동 모델을 제안하며, 연결, 성능, 장애, 구성관리 별로 연동 시나리오를 제시하였다. 본 논문에서 제안된 연동 모델은 GSMP 마스터와 SNMP 에이전트가 컨트롤러에서 연동하는 모델이다.

추후 연구사항으로는 MPLS-LSR MIB만을 대상으로 한 연동 시나리오를 MPLS-FTN MIB로 확장하여 연동 시나리오를 제시하는 것이다.

본 논문은 한국과학재단 우수연구센터사업의 연구 결과입니다.

참고 문헌

- [1] A. Doria, et al., "General Switch Management Protocol," Internet draft, <draft-ietf-gsmp-09.txt>, May 2001.
- [2] 이성창, "고속 인터넷 서비스를 위한 MPLS," 대한전자공학회 텔레콤, 제15권 1호, 1999년 6월.
- [3] Jeremy Lawrence, "Designing Multiprotocol Label Switching Networks," IEEE Communications Magazine, July 2001.
- [4] E. Rosen et al., "Multiprotocol Label Switching Architecture," RFC 3031, January 2001.
- [5] Thomas D. Nadeau et al., "Multiprotocol Label Switching (MPLS) FEC-ToNHLFE (FTN) Management Information Base Using SMIPv2," Internet draft, <draft-ietf-mpls-ftn-mib-01.txt>, April 2001.
- [6] Cheenu Srinivasan et al., "MPLS Label Switch Router Management Information Base Using SMIPv2," Internet draft, <draft-ietf-mpls-lsr-mib-07.txt>, January 2001.
- [7] Nils Bjorkman et al., "The Movement from Monoliths to Component-Based Network Elements," IEEE Communications Magazine, January 2001.
- [8] 차 영욱, "Toward Open Architecture in the Optical Internet," 제 3회 광 인터넷 워크샵, 2001년 6월.
- [9] AdventNet, "AdventNet Agent Toolkit (java Edition) 4.2 Documentation," August 2001.