ISBN 978-89-6211-687-8

오픈플로우 기반의 QoS 라우팅 기술

일자: 2010 년 11 월 11 일

부서: 슈퍼컴퓨팅본부 융합자원실

제출자: 홍원택, 권윤주

{wthong, yulli}@kisti.re.kr

한국과학기술정보연구원
Korea Institute of Science and Technology Information
305-806 대전광역시 유성구 어은동 52번지
TEL (042)869-0676 / FAX (042)869-0679
www.kisti.re.kr

목차

1. 서론

2. 오픈플로우 적용 환경

3. QoS 플로우 라우팅 전체 시스템 구조

4. 구성 정보를 위한 데이터베이스 스키마

5. 주요 컴포넌트간 흐름도

6. 사용자 인터페이스 사용 방법

7. 결론

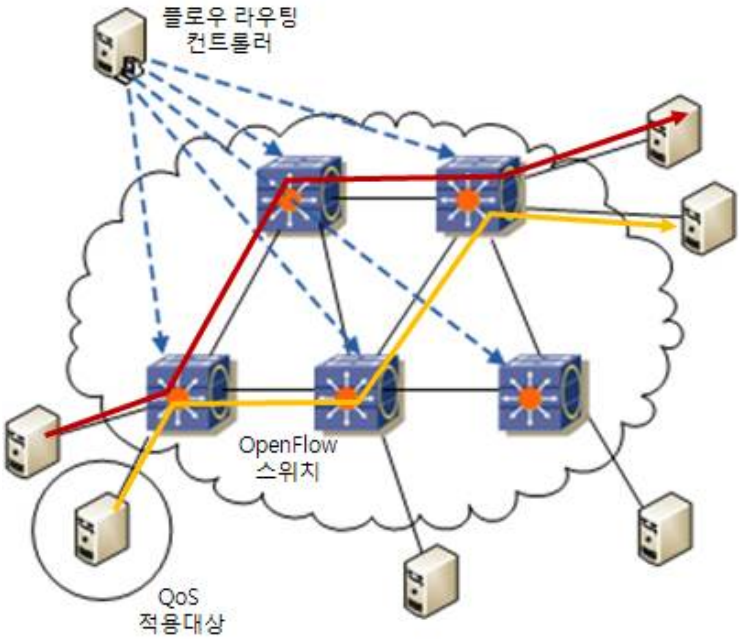
8. 참고문헌

1. 서론

OpenFlow 기술은 Stanford 대학의 Clean slate 프로그램에서 연구 진행하는 공개된 표준 프로젝트로서, 실험적인 네트워크 프로토콜들을 실제 네트워크 상에서 구동하도록 해줌으로써, 가상화된 프로그래머블한 네트워크의 구성을 가능하게 해 준다. 이러한 기술을 통해 사용자는 원하는 시나리오에 따른 네트워크 운영 기술을 개발할 수 있고, 플로우 기반의 데이터 평면과 제어 평면의 분리를 통해 관리의 유연함을 도모할 수 있게 된다. OpenFlow 기술의 특성 상 다양한 형태의 플로우 라우팅이 가능할 수 있고, 이러한 아이디어들은 NOX 와 같은 컨트롤러 상에 플러그인 형태로 반영될 수 있다.
본 문서에서는 OpenFlow 네트워크에서 트래픽 부하를 고려한 동적 플로우 라우팅 기술 및 사용자 인터페이스를 사용하는 방법을 기술한다. 본 기술을 통해 사용자가 요구하는 서비스 품질에 따라 동적으로 트래픽 전송 경로를 할당하고, 네트워크 상황을 인지하여 플로우의 재정의를 통해 트래픽을 이동시킴으로써 사용자의 트래픽을 관리 및 제어할 수 있다. 결과적으로 일반 플로우와 특정 임계치를 보장받고 싶어하는 QoS 플로우를 구분함으로써 특정 네트워크 응용을 지원하기 수월해 질 수 있고, 동적인 플로우 이동을 통해 유휴 대역을 효율적으로 이용할 수 있게 됨으로써 망 이용의 효율성 측면에 도움이 될 것이다.

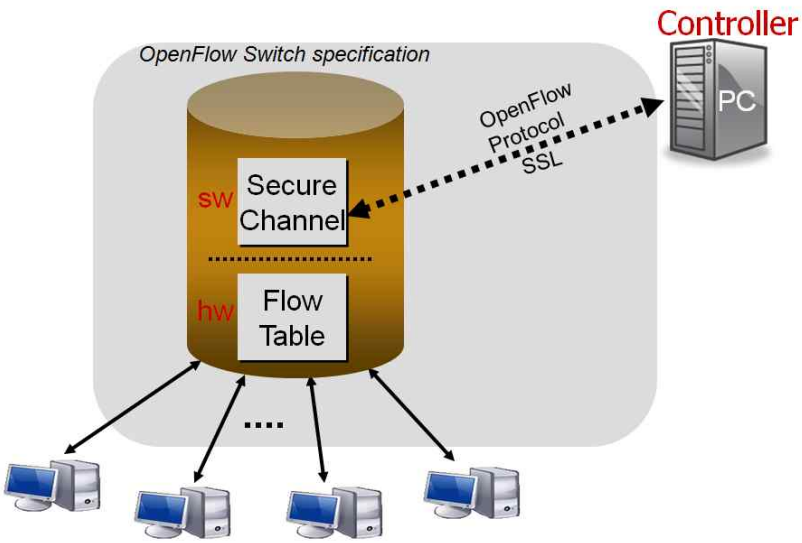
2. 오픈플로우 적용 환경

본 절에서는 QoS 플로우 라우팅 기술을 적용하기 위한 환경을 살펴본다. 적용 환경은 아래 그림과 같이 OpenFlow 스위치들과 컨트롤러로 구성되고, 플로우 라우팅 컴포넌트는 컨트롤러 상에 위치하게 된다.



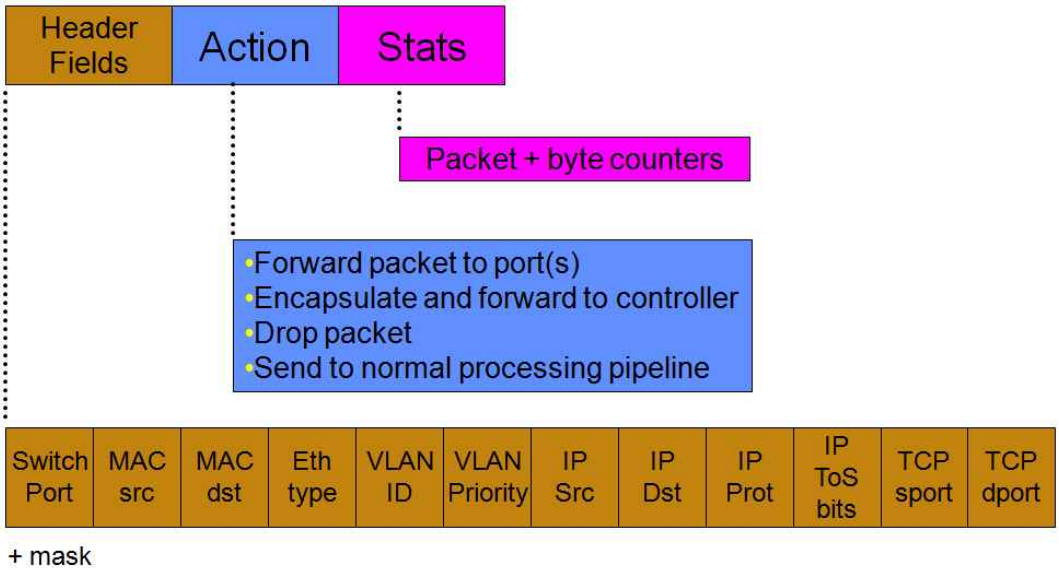
<그림 1> OpenFlow 스위치 및 컨트롤러로 구성된 플로우 라우팅 시스템 적용 환경

서론에서 언급되었듯이, OpenFlow 기술은 기존의 정적 네트워크 환경을 개선하여, 사용자의 요구를 반영하는 사용자 정의 네트워크 환경을 제공해 줄 수 있는 근간이 되는 공개된 표준이다. OpenFlow 네트워크의 기본 환경은 아래 그림과 같이 OpenFlow 스위치와 컨트롤러로 구성되어 있고, 이들은 TCP 또는 SSL 과 같은 통신방법을 통해 정보를 교환한다.



<그림 2> OpenFlow 스위치와 컨트롤러의 관계

송신측 노드에서 연결하고자 하는 수신측 노드로 통신을 요청하면 OpenFlow 스위치는 송신측으로부터 받은 데이터를 버퍼에 저장한 후, 컨트롤러에게 이 플로우에 대한 정보를 전송한다. 플로우에 대한 정보 교환은 OpenFlow 프로토콜을 이용하여 이루어진다. 컨트롤러에서는 받은 플로우 정보와 사용자의 요구를 반영하여 플로우에 대한 포워딩(액션)을 생성하고, 아래 그림과 같은 플로우 엔트리의 형태로 OpenFlow 스위치에 내려준다. 플로우 엔트리를 받은 OpenFlow 스위치는 엔트리 안에 포함된 포워딩(액션) 규칙에 기반하여 이 후의 트래픽을 관리한다. 플로우 엔트리는 목적지/발신지 MAC, IP, 서비스 포트 번호 등 10 가지 튜플들로 형성되는 헤더 필드와 지정된 포트로 포워딩을 수행하는 액션 필드 등으로 구성된다.

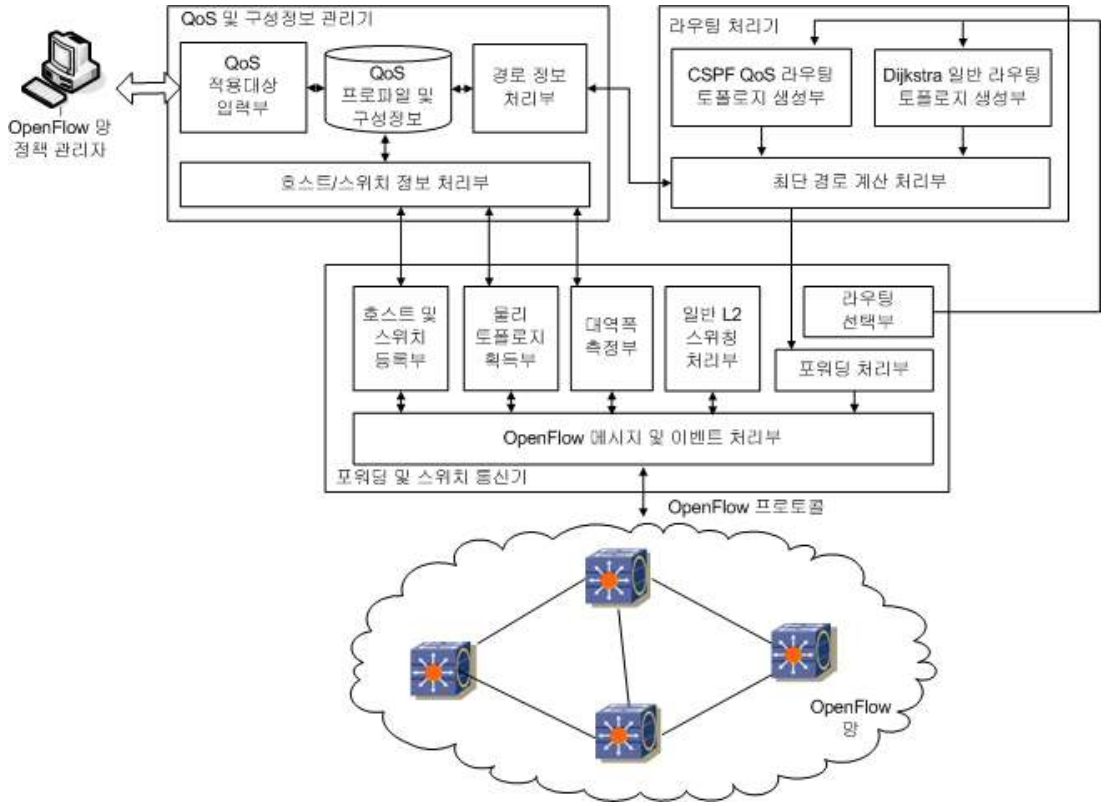


<그림 3> OpenFlow 프로토콜의 플로우 엔트리 구조

QoS 플로우 라우팅 기술은 이와 같은 OpenFlow 네트워크 환경에서 일반 플로우와 망 정책 관리자에 의해 정의되어 차별화된 요구 사항을 보장하는 QoS 플로우를 구별하여 라우팅을 할 수 있는 시스템으로서, 이 후에서는 컨트롤러 상에 구현되는 라우팅 시스템, QoS 및 네트워크 구성 정보를 정의하는 데이터베이스 스키마, 플로우 라우팅을 위한 알고리즘, 테스트를 위한 도구 등을 기술한다.

3. QoS 플로우 라우팅 전체 시스템 구조

아래 그림은 트래픽 부하를 고려한 동적 QoS 플로우 라우팅 시스템의 전체 구조를 상세히 보여준다. 시스템은 QoS 및 구성정보 관리기, 포워딩 및 스위치 통신기, 라우팅 처리기로 구성되어 있다.



<그림 4> 동적 QoS 플로우 라우팅 시스템 전체 구조

QoS 및 구성정보 관리기는 OpenFlow 네트워크 정책 관리자로부터 입력받은 QoS 정책, 네트워크 연결 상태를 유지하는 네트워크 구성 정보로 구성된 QoS 프로파일 및 구성 정보 데이터베이스를 중심으로 구성된다. 그리고, 이 데이터베이스의 테이블을 생성/유지/업데이트하기 위해 각각 QoS 적용대상 입력부, 호스트/스위치 정보 처리부, 경로 정보 처리부 등으로 구성된다. 핵심이 되는 데이터베이스의 스키마 구조 및 필드에 대한 설명은 다음 절에서 설명된다.

포워딩 및 스위치 통신기는 본 시스템의 가장 중추적인 역할을 수행하는 부분이다. OpenFlow 프로토콜에 대한 전반적인 대응을 수행하는 OpenFlow 메시지 및 이벤트 처리부, datapath 이벤트에 기반하여 컨트롤러에 감지된 OpenFlow 스위치와 호스트를

등록하는 호스트 및 스위치 등록부, LLDP 패킷을 기반으로 물리 토폴로지를 인식하는 물리 토폴로지 획득부, port_stats 를 이용하여 노드 간의 대역을 얻어오는 대역폭 측정부, 기존의 L2 스위칭 역할을 수행하는 일반 L2 스위칭 처리부, 3 계층 이상의 패킷에 대해 선언된 QoS 프로파일에 기반하여 라우팅 결정을 내리는 라우팅 선택부, 설정된 경로에 따라 포함되는 OpenFlow 스위치로 플로우 엔트리를 생성하여 전송하는 포워딩 처리부로 구성된다.

라우팅 처리기는 사용자의 네트워크 요구 사항을 반영하는 플로우에 대한 처리를 담당하는 CSPF QoS 라우팅 토폴로지 생성부, 그 외 일반 플로우 처리를 담당하는 Dijkstra 일반 라우팅 토폴로지 생성부로 구성된다. 그리고, 각각 생성된 토폴로지에 기초하여 최단 경로를 발견해 주고, 네트워크 구성정보 데이터베이스에 반영을 요청하는 최단 경로 계산 처리부로 구성된다.

4. 구성 정보를 위한 데이터 스키마

본 절에서는 본 시스템에서 정의하는 데이터베이스의 스키마 구조를 설명한다. 정의된 스키마들을 기반으로 QoS 프로파일 및 구성정보 데이터베이스 생성/관리한다. 정의된 스키마들은 OpenFlow Switch 연결정보, OpenFlow Switch 리스트, Host 정보, QoS 정책, Dijkstra Path 정보, QoS Path 정보, Fail 패킷 정보 등으로 구성된다.

OpenFlow Switch 연결정보는 OpenFlow 스위치가 다른 OpenFlow 스위치와 연결된 상태 정보를 나타내고, OpenFlow 스위치마다 하나의 테이블을 갖게 된다. PT 는 해당 스위치가 다른 스위치와 붙은 포트번호, COF 는 해당 스위치가 연결된 다른 OpenFlow 스위치의 ID, RBW 는 수신측 대역폭, TBW 는 송신측 대역폭, CRT 는 마지막으로 LLDP 패킷을 받은 시간으로 연결 상태를 주기적으로 확인하기 위함이다. CST 는 링크의 cost 로 기본 값은 1 이다. OpenFlow Switch 리스트는 OpenFlow 망에 가입된 스위치의 ID 를 저장한다.

PT	COF	RBW	TBW	CRT	CST

<그림 5> OpenFlow Switch 연결정보

DPD

<그림 6> OpenFlow Switch 리스트

Host 정보는 OpenFlow 망에 가입된 호스트의 정보를 저장하는 용도이다. MAC 는 연결된 호스트의 MAC 주소, IP 는 IP 주소, COF 는 해당 호스트가 연결된 OpenFlow 스위치의 ID, CPT 는 연결된 스위치의 포트번호, RBW 는 수신측 대역폭, TBW 는 송신측 대역폭, CRT 는 호스트가 연결된 시간, USE 는 경로 설정 시 사용 여부를 나타낸다.

MAC	IP	COF	CPT	RBW	TBW	CRT	USE

<그림 7> Host 정보

QoS 정책은 OpenFlow 망 정책 관리자로부터 입력받은 QoS 적용 관련 정보를 저장하는 용도로 사용되고, 와일드 카드(*)를 지원한다. SM, DM 은 각각 송수신 측의 MAC 주소, SP, DP 는 각각 송수신 측의 IP 주소, Sp, Dp 는 각각 송수신 측의 서비스 포트 번호, DLT 는 데이터링크 계층에서의 타입, NWP 는 네트워크 프로토콜 번호, PRT 는 우선순위, TYP 는 QoS 플로우 경로 설정 시 사용되는 라우팅 프로토콜 옵션으로 본 시스템에서는 CSPF 를 이용한다. CT 는 제약조건의 만족시키고자 하는 임계치 값으로 주로 대역폭 정보를 이용한다. DRT 는 QoS 플로우의 단/양방향 여부, STE 는 해당 엔트리의 사용 상태를 나타낸다.

SM	DM	SP	DP	Sp	Dp	DLT	NWP	PRT	TYP	CT	DRT	STE

<그림 8> QoS 정책

Dijkstra Path 정보는 QoS 보장을 필요로 하지 않는 일반 플로우의 경로 정보를 저장하는 용도로 사용된다. SM, DM 은 각각 송수신 측의 MAC 주소, SP, DP 는 각각 송수신 측의 IP 주소, Sp, Dp 는 각각 송수신 측의 서비스 포트 번호, DLT 는 데이터링크 계층에서의 타입, NWP 는 네트워크 프로토콜 번호를 나타낸다. PTH 는 해당 플로우가 거쳐 가야 하는 스위치들의 ID 를 순차적으로 포함한다. USE 는 해당 경로의 사용 여부를 나타내고, EXP 는 해당 경로가 마지막으로 사용된 시간을 기록한다.

SM	DM	SP	DP	Sp	Dp	DLT	NWP	PTH	USE	EXP

<그림 9> Dijkstra Path 정보

QoS Path 정보는 QoS 플로우의 경로에 대한 정보를 저장하는 용도로 사용되고, 일반 플로우의 Path 정보 스키마와 상당 부분 유사하다. SM, DM 은 각각 송수신 측의 MAC 주소, SP, DP 는 각각 송수신 측의 IP 주소, Sp, Dp 는 각각 송수신 측의 서비스 포트 번호, DLT 는 데이터링크 계층에서의 타입, NWP 는 네트워크 프로토콜 번호, PRT 는 우선순위, TYP 는 QoS 플로우 경로 설정 시 사용되는 라우팅 프로토콜 옵션으로 본 시스템에서는 CSPF 를 이용한다. CT 는 제약조건의 만족시키고자 하는 임계치 값으로 주로 대역폭 정보를 이용한다. PTH 는 해당 플로우가 거쳐 가야 하는 스위치들의 ID 를 순차적으로 포함한다. DRT 는 QoS 플로우의 단/양방향 여부, USE 는 경로 설정 시 사용 여부, EXP 는 해당 경로가 마지막으로 사용된 시간을 기록한다. ABW 는 이 경로의 가용 대역폭을 나타내고, CPT 는 경로를 구성하는 홉 간의 링크들의 측정 여부를 리스트 형태로 나타낸다. CBW 는 여러 링크들 중에 가장 작은 가용 대역폭을 임시로 저장하는 용도이며, CAY 는 해당 경로가 다른 경로에 의해 첨범 당했는지를 나타낸다.

SM	DM	SP	DP	Sp	Dp	DLT	NWP	PRT	TYP	CT	PTH	DRT	USE	EXP	ABW	CPT	CBW	CAY

<그림 10> QoS Path 정보

Fail 패킷 정보는 경로 설정 실패 시에 실패한 패킷에 대한 정보를 저장하는 용도로 사용된다. Path 정보를 제공하는 스키마의 구조와 유사하며, CRT 값으로 경로 설정 실패 시의 시간을 기록한다.

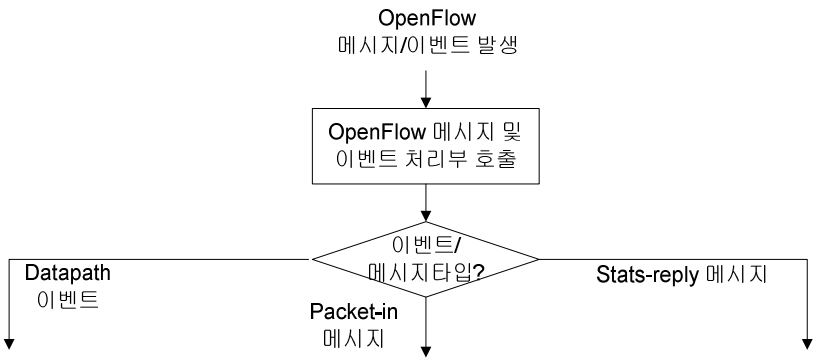
SM	DM	SP	DP	Sp	Dp	DLT	NWP	CRT

<그림 11> Fail 패킷 정보

5. 주요 컴포넌트간 흐름도

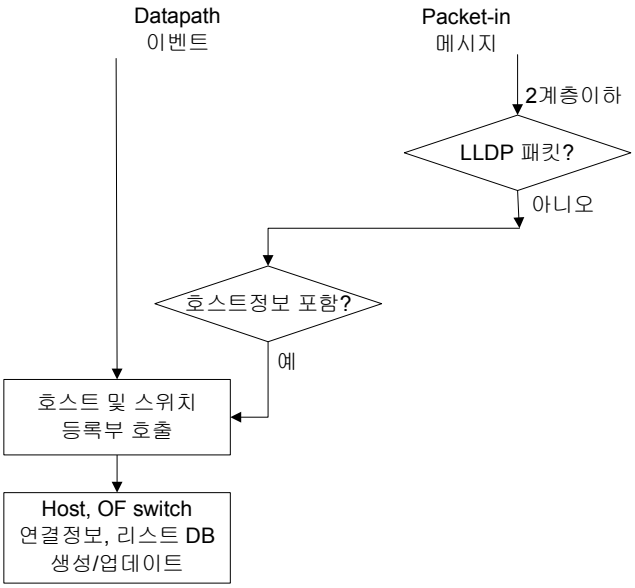
본 절에서는 컨트롤러에서 OpenFlow 메시지/이벤트 수신 시 포워딩 및 스위치 통신기에서 각 메시지 타입과 이벤트에 따라 처리하는 단계 별 알고리즘을 기술한다.

아래 그림은 본 시스템에서 관장하는 OpenFlow 스위치로부터 메시지 또는 이벤트가 발생할 경우 처리 단계를 보여준다. OpenFlow 메시지 또는 이벤트가 발생하면 이벤트 또는 메시지의 타입에 따라 대응하게 된다. 이러한 대응은 OpenFlow 메시지 및 이벤트 처리부를 호출하여 수행되고, OpenFlow 프로토콜에서 정의된 여러 종류의 메시지 및 타입들 중에서 본 시스템에서는 스위치의 가입/탈퇴를 알리는 Datapath 이벤트, 스위치에서 컨트롤롤러로 전송되는 Packet-in 메시지, 스위치의 관리 정보 요구를 위한 Stats-request 에 대한 Stats-reply 에 대해 처리한다.



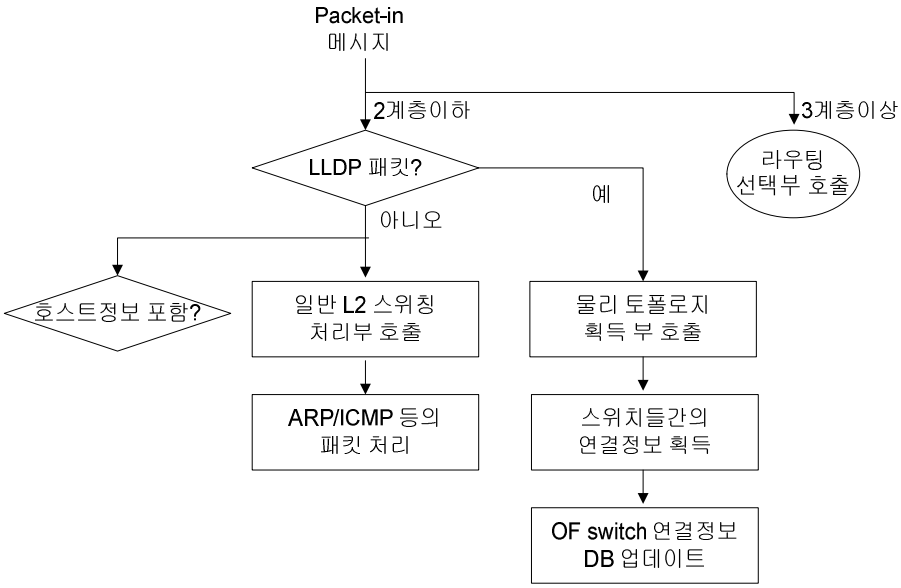
<그림 12> OpenFlow 메시지/이벤트 발생시 분류

Datapath 이벤트의 발생은 OpenFlow 스위치가 컨트롤러의 영역에 가입/탈퇴를 한다는 것을 의미하므로, 호스트 및 스위치 등록부에서 OpenFlow Switch 리스트 데이터베이스에 엔트리를 추가/삭제하고, 해당 스위치를 위한 OpenFlow Switch 연결정보 테이블을 생성하고 각 포트별로 엔트리를 생성 및 추가/삭제한다.



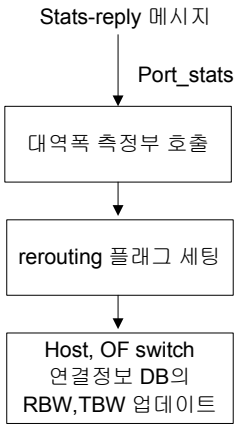
<그림 13> Datapath 이벤트 발생 시 처리

Packet-in 메시지의 경우에는 여러 종류의 메시지가 캡슐화 되어 컨트롤러로 전달되게 되는데, 크게 나누어 2 계층 이하의 패킷과 3 계층 이상의 정보를 포함하는 패킷으로 나누어 처리한다. 2 계층 이하의 패킷들 중에 ARP 와 같은 목적지 주소를 모르는 패킷, ICMP 등의 패킷들은 범용적으로 사용되는 스위치와 같은 방식으로 동작을 하게 되어 있고, 이러한 패킷들을 처리하면서 호스트 정보가 포함된 경우에는 Host 정보 테이블을 생성하여 유지하게 된다. 그리고, LLDP 패킷을 수신하는 경우에는 물리 토폴로지 획득 부를 호출하여 스위치들 간의 연결 정보를 획득하게 된다. 주요 사항으로 LLDP 패킷의 chassid 와 portid 는 최초 LLDP 패킷을 받은 스위치의 정보를 나타내며 dpid 와 inport 는 최초 스위치에서 보낸 LLDP 패킷을 받은 스위치의 정보를 나타내므로, 이러한 정보를 OpenFlow Switch 연결정보 테이블에 저장하고 경로 설정 시 다시 불러와 사용한다. 추가적으로 호스트의 정보를 얻기 위해 호스트가 어느 스위치의 몇 번 포트에 연결되어 있는지 감지하여 Host 정보 테이블에 저장하고 변동 시 데이터베이스를 갱신 해 준다.



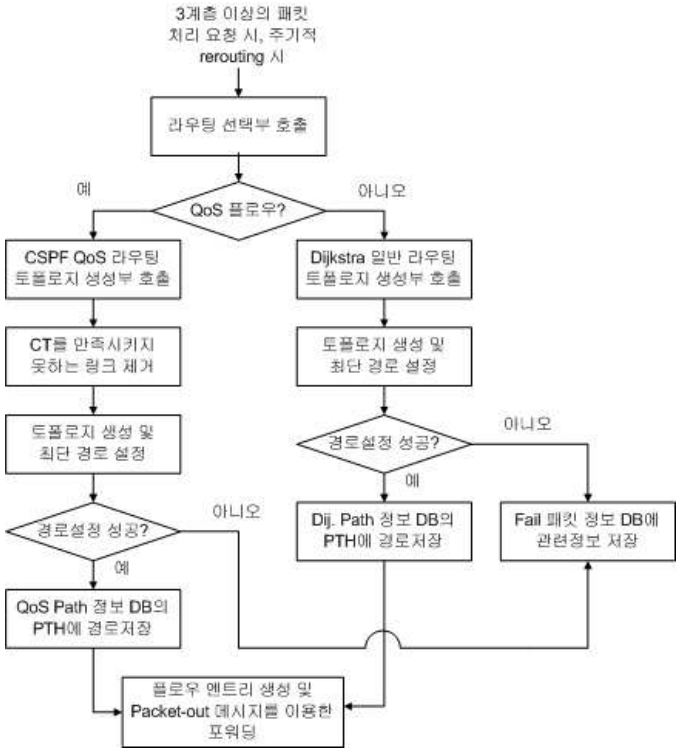
<그림 14> Packet-in 메시지 수신 시 처리

Stats-reply 메시지는 스위치의 포트 별 통계 자료를 수집하기 위해 주기적으로 수신한다. 메시지 수신 시 포함된 포트별 통계 정보를 이용하여 링크 간의 대역폭을 계산한다. 대역폭 계산 방법은 이전에 보고 받은 누적 바이트 수와 시간을 각각 현재 보고받은 누적 바이트 수와 시간의 차를 계산한 후 바이트 차이를 시간 차이로 나눈다. 이 값은 OpenFlow Switch 연결정보 테이블 및 Host 정보 테이블의 RBW, TBW 에 저장되며 이 값은 각 링크의 사용 대역폭이 된다. QoS 경로 상 홉 간의 가용 대역폭 계산을 위해서는 최대 지원 대역폭을 사용 대역폭으로 빼주어 계산한다. 그러나 이 값은 경로 상의 여러 링크 중 하나의 값이므로 그 경로의 가용 대역폭이라 할 수 없다. 경로의 가용 대역폭은 관련 링크들 중 최소 가용 대역폭이 되어야 하기 때문이다. 따라서 각 링크의 가용 대역폭을 전부 체크한 후 그 중 최소 가용 대역폭을 도출해야 한다. QoS Path 정보 테이블의 CBW 는 현재까지 체크한 링크들의 가용 대역폭 중 가장 최소값을 임시로 저장하고 있다. CPT 는 경로에 포함된 각 링크의 가용 대역폭 체크상태를 나타낸다. 경로에 포함된 링크가 5 개 이면 ‘11111’이 체크되어야 하고 2 개 이면 ‘11’이 된다. 링크의 순서에 따라 ‘1'이 체크되며 5 개 링크 중 첫 번째와 세 번째 링크가 체크되었으면 ’10100‘이 된다. 포함된 모든 링크의 가용 대역폭이 체크 완료되면(즉, 모두 1) 그때 CBW 는 ABW 로 저장된다.



<그림 15> Stats-reply 메시지 수신 시 처리

아래 그림은 3 계층 이상의 정보를 포함하는 Packet-in 메시지 수신 시와 재 라우팅이 요구되는 시점에 라우팅 선택부가 호출되는 경우를 설명한다. 본 시스템에서는 일반 플로우와 QoS 플로우를 나누어 처리하는데, QoS 플로우인 경우, CSPF QoS 라우팅 토폴로지 생성부를 호출하여 OpenFlow Switch 연결 정보 테이블로부터 제약사항(CT)을 만족시키지 못하는 자원을 없앤 후 새로운 토폴로지를 만든다. 이 후, 최단 경로 설정을 통해 경로를 생성하고, 성공하면 QoS Path 정보 테이블의 PTH 필드에 해당 경로를 리스트 형태로 저장한다. 반면, 일반 플로우의 경우에는 제약조건 없이, 단순하게 토폴로지를 생성한 후 최단 경로를 생성하여 성공하면 Dijkstra Path 정보 테이블 PTH 필드에 해당 경로를 리스트 형태로 저장한다. 이후, 경로에 포함된 OpenFlow 스위치에 각각 필요한 플로우 엔트리를 생성하고, OpenFlow 의 Packet-out 메시지를 이용하여 포워딩 함으로써 경로상의 스위치마다 필요한 포워딩을 수행할 수 있게 된다.



<그림 16> 라우팅 처리기에서의 플로우에 따른 처리 알고리즘

6. 사용자 인터페이스 사용 방법

OpenFlow 기반의 QoS 라우팅 시스템을 적용하고, 테스트하기 위해서는 몇 가지 설정 정보들이 추가적으로 요구되고, 일반 라우팅 및 QoS 라우팅 등의 중간 과정을 확인할 필요가 있다. 이러한 설정과 관련된 파일로는 QoS 플로우에 대한 정의를 하는 QoS 프로파일과 라우팅과 관련된 연결 상태 및 모니터링 등의 주기 등을 설정하는 parameter 파일 등이 존재한다. 또한 라우팅 경로 등의 중간 과정을 확인하기 위해 Sqlite-Browser 와 같은 데이터베이스 브라우저를 사용한다. 본 절에서는 관련된 일련의 과정들을 기술한다.

가. QoS 파일

- 용도: 원하는 특정 플로우에 QoS 를 지원하기 위한 파일

- 사용법:

(1) 파일의 가장 윗줄에 '#'으로 주석처리 되어있는 부분이 QoS 파일에 입력해야 하는 필드들이다. 주석에는 각 필드의 명칭이 나와 있다. 정의된 필드는 구성 정보를 위한 QoS 정책 데이터베이스 스키마와 일치한다.

(2) MAC, IP, port, vlan_id, vlan_pcp, dl_type, nw_proto 는 플로우를 정의하는 10 튜플이며, priority, type 은 확장을 위해 정의하였으나, 현재 사용하지 않으며 type 의 경우에는 'CSPF'를 써 주어야 한다. direct 는 QoS 플로우가 양방향 연결인지 단방향 연결인지 정의해주며 'sim'(단방향)과 'du'(양방향)가 있다. state 는 'used', 'add', 'mod', 'del'가 있으며 처음 값을 입력할 경우 앞의 3 가지 중 선택하면 된다. 이전에 입력된 값을 지우고 싶을 경우는 state 항목에 'del'을 써준다.

(3) 필수적으로 입력해 주어야 하는 항목은 type(CSPF), direct, state, ct 와 하나 이상의 플로우 주소이다. 그 외에는 대소문자 x 를 넣어주면 된다. 다음은 작성 예이다.

x/x/10.0.0.32/x/x/x/x/x/x/x/1/CSPF/500000000/du/used

(4) 어플리케이션에서 사용 중인 경우에 불러오거나 저장 시, 경고메시지가 나올 수 있지만, 무시하고 저장하거나 불러오면 된다.

나. parameter 파일

- 용도: QoS 라우팅 어플리케이션 실행에 필요한 설정 값을 변경하는 파일

- 사용법:

(1) '#'은 주석처리를 위해 쓰인다.

(2) 시간단위는 초이며, 대역폭 단위는 bits/sec 이다.

(3) QoS_Reading_Period 는 QoS 파일을 얼마나 자주 읽어 들일지 설정하는 부분이다.

(4) RErouting_period 는 '경로 재설정 함수'의 실행 주기를 설정하는 부분이다.

(5) PORT_LIVE_PERIOD 는 OpenFlow 스위치들끼리 연결된 포트가 사용 중인지 여부를 판단하는 시간을 설정한다.

(6) DB_CHECK_PERIOD 는 유지하고 있는 구성 정보 데이터베이스를 주기적으로 체크하는 시간을 설정한다.

(7) MIN_BW 는 MAX_AV_BW 를 1G 로 잡았을 때 실제 가용량이 900M 라면 100M 으로 설정해 주어야 한다. 망이 스팩 상으로 1G 망이라 하더라도 실제 사용중인 경우에 대역폭이 적게 나올 경우가 있는데 이것을 조절하기 위해 이 두 파라미터가 사용된다. 즉, 실제 대역폭은 'MAX_AV_BW - MIN_BW'로 설정된다. 그래서 MIN_BW 는 QoS 경로가 다른 것에 침범 당했는지 아닌지 결정하는데 사용된다.

(8) MAX_AV_BW 는 스팩 상 최대 대역폭으로 설정한다.

(9) RR_M_BW 은 대역폭을 가용 대역폭의 30%이상 변화할 때에만 변경한다. 그러나 가용대역폭이 작아 그 30%가 빈번하게 변할 경우를 막기 위해 변화량이 최소 RR_M_BW 이상 되어야 데이터베이스에 저장한다.

(10) PARA_PERIOD 는 이 파일을 주기적으로 불러오는 시간 설정한다.

(11) CACHE_TIMEOUT 은 플로우 테이블의 'idle_timeout' 값을 설정한다.

(12) MONITOR_PORT_PERIOD 는 포트 통계 자료 요청 메시지를 보내는 주기를 설정한다.

(13) parameter 파일 역시 QoS 파일처럼 어플리케이션에서 사용 중인 경우, 강제로 저장하거나 불러오면 된다.

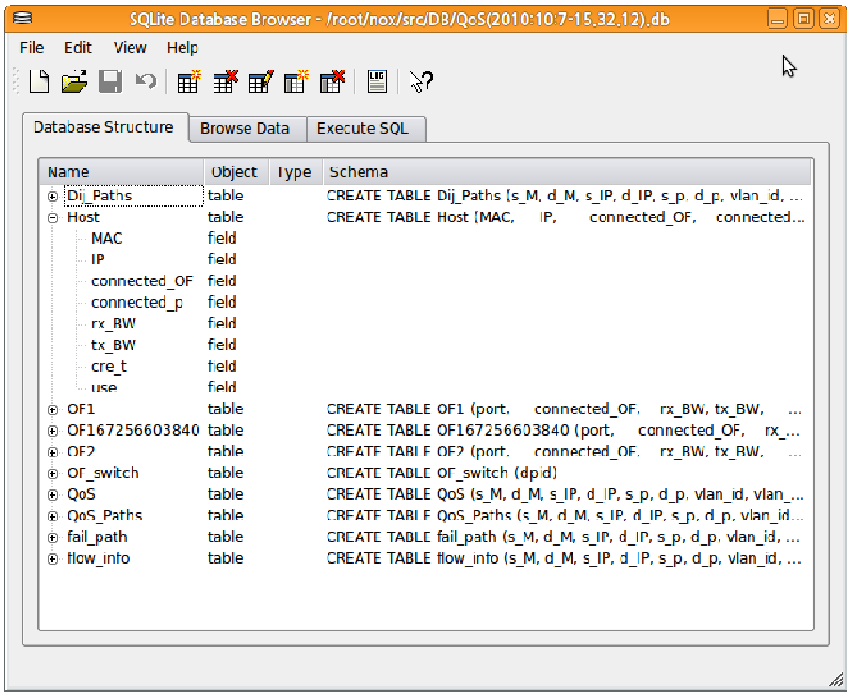
다. Sqlite-Browser

- 실행: 터미널 창에서 'sqlitebrowser'를 입력

- 사용법:

(1) 메뉴의 'file'->'open database'에서 원하는 데이터베이스 파일을 선택한다.

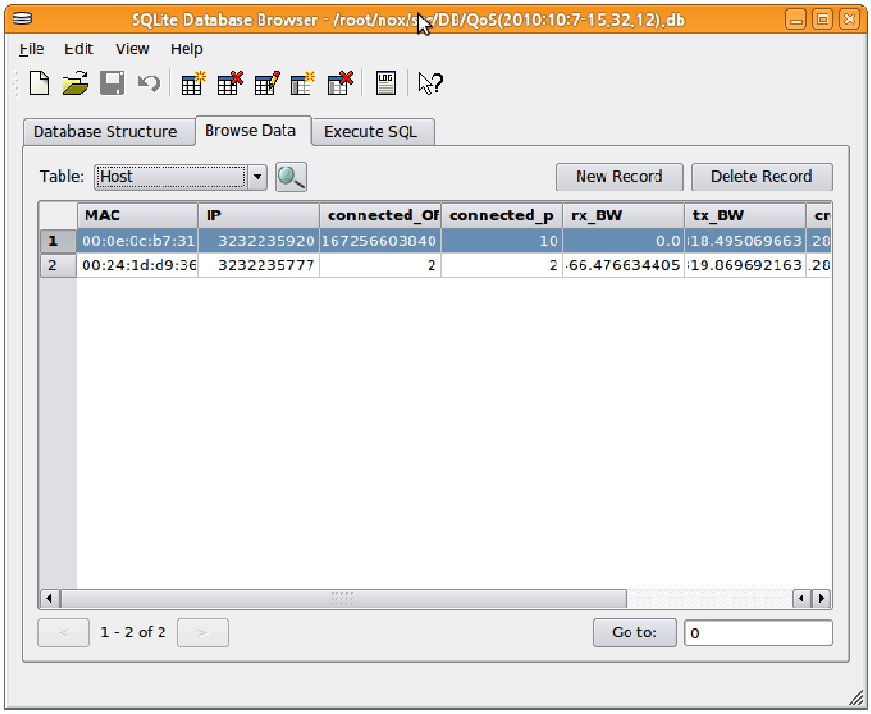
(2) 선택 파일을 열면 다음과 같은 창이 나온다.



<그림 17> SQLite 브라우저 실행 화면

위의 탭 중에서 'database structure'는 데이터베이스의 구조를 보여준다. 가장 상위 이름은 테이블의 이름이며 그 하위 이름은 각 필드의 이름이다.

(3) 'browse data' 탭은 아래 그림과 같으며 테이블 별로 구체적인 자료를 보여준다.

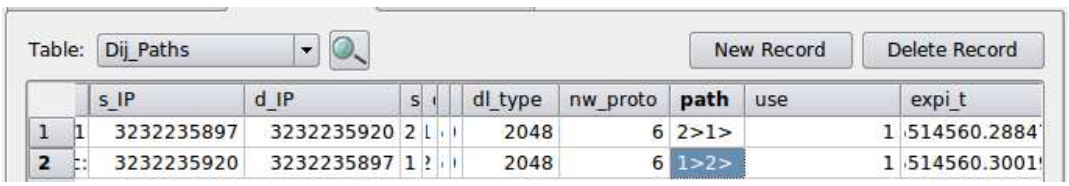


<그림 18> Host 테이블의 상세 정보

'Table:' 옆의 화살표로 원하는 테이블을 선택하여 정보를 볼 수 있다. 각 정보는 위 그림처럼 각 필드로 나뉘어 있으며 정보를 순차적으로 표시하고 있다.

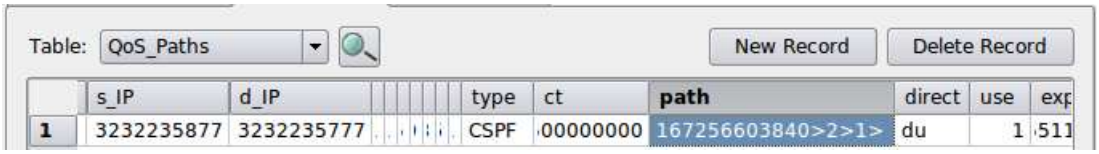
(4) 주의할 점으로 이 데이터베이스 파일을 변경하거나 삭제해서는 안 된다. 단지, 데이터베이스의 내용을 보는 목적으로 사용한다.

(5) 아래 그림은 Dij_Paths 테이블 내에서 포함하는 상세정보를 나타낸다. 여러 가지 필드 중에서 'path' 필드를 통해서 거쳐가는 OpenFlow 스위치들의 경로를 추적할 수 있다.



<그림 19> Dij_Paths 테이블 내의 경로 정보 표시

(6) 아래 그림은 QoS_Paths 테이블 내에서 포함하는 상세정보를 나타낸다. Dij_Paths 테이블에 존재하는 필드들 외에도 데이터베이스 스키마에 정의되어 있듯이 QoS 플로우의 단/양방향 여부를 나타내는 'direct' 필드도 볼 수 있다.



<그림 20> QoS_Paths 테이블 내의 경로 정보 표시

7. 결론

OpenFlow 스위칭 기술은 사용자 정의 네트워크 시나리오를 작성하여, 플러그인 형태로 반영하여 실제 네트워크 상에서 운영할 수 있는 기술로 자리잡아 가고 있다. 최근에는 주요 네트워크 벤더들도 OpenFlow 스펙을 장비들에 반영하여 출시하고 있는 상황이다. 이러한 기술은 OpenFlow 스위치 중심의 데이터 평면과 NOX 컨트롤러 중심의 제어 평면을 분리함으로써 네트워크 운영의 유연성을 높여줄 수 있다.

본 문서에서는 OpenFlow 망에서 트래픽 부하를 동적으로 반영하는 QoS 플로우 라우팅 시스템 및 사용자 인터페이스를 기술하였다. 세부적으로는 시스템의 적용 환경, 플로우 라우팅 시스템의 전체 구조 및 컴포넌트간 관계도, 네트워크 구성 정보를 제공하기 위한 데이터베이스 스키마 등을 포함하였다. 본 시스템을 통해 일반 플로우 및 일정 수준 이상의 서비스 품질을 보장받고자 하는 특정 플로우를 분류하여 처리할 수 있으며, 주기적인 망의 상황을 모니터링 함으로써 OpenFlow 망의 진입 시점뿐만 아니라, 운영 중에도 전체적인 망의 상황에 따라 플로우 이동을 함으로써 능동적인 대응이 가능할 것으로 기대된다.

8. 참고문헌

[1] OpenFlow, http://www.openflowswitch.org/

[2] OpenFlow Switch Specification 0.8.9, http://www.openflowswitch.org

[3] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. OpenFlow: enabling innovation in campus networks, ACM SIGCOMM Computer Communication Review, 38(2):69–74, April 2008

[4] NOX controller, http://www.noxrepo.org/

[5] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, NOX: Towards an operating system for networks, ACM SIGCOMM Computer Communication Review, July 2008

[6] GENI: Global Environment for Network Innovations, http://www.geni.net/

[7] Larry Peterson, John Wroclawski, "Overview of the GENI architecture", GDD-06-11, January 5, 2007.

[8] R.Guerin and A.Orda, "QoS-based Routing in Networks with Inaccurate Information: Theory and Algorithms." IEEE INFOCOM ' 97, Japan, Apr. 1997

[9] I.Cidon, R.Rom and Y.Shavitt, "Multi-Path Routing Combined with Resource Reservation," IEEE INFOCOM ' 97, Japan, Apr. 1997, pp. 92-100

[10] Ziegelmann and Mark, "Constrained Shortest Paths and Related Problems: Constrained Network Optimization", VDM Verlag Dr. Müller. ISBN 978-3-8364-4633-4. December 2007