OpenFlow 기반 네트워킹 실험을 위한 플로우 제어 인터페이스 구현

신성호, 김남곤, 김종원
광주과학기술원 정보통신공학과 네트워크미디어 연구실

Implementing a Flow Control Interface for OpenFlow-based Networking Experiments

Sungho Shin , Namgon Kim , and JongWon Kim
Networked Media Laboratory, Dept. of Information and Communications,
Gwangju Institute of Science and Technology (GIST)
{shshin, ngkim, jongwon}@nm.gist.ac.kr

요 약

OpenFlow 는 인터넷 스위치(라우터 포함)를 부분적으로 개선하여 연구자가 기존 네트워크의 정상적인 서비스를 간섭하지 않으면서 새로운 네트워킹 기술을 테스트할 수 있는 미래인터넷 (Future Internet) 연구를 위한 프로토콜이다. 하지만 OpenFlow 기반 테스트베드의 상태 정보를 확인하고 발생한 플로우들을 제어할 수 있는 도구(tool)가 부족하여, 연구자가 실험의 다양성을 제공 받기 어렵다. ENVI(Extensible Network Visualization & Control Framework)는 OpenFlow 네트워크 가시화를 위한 프레임워크로, 사용자의 목적에 맞는 인터페이스를 만들도록 제공해 준다. 본 논문에서는 OpenFlow 기반 테스트베드의 토폴로지 정보 및 플로우 상태를 쉽게 확인하고, 플로우 경로를 제어하는 OpenFlow 제어기 응용의 개발과 ENVI 의 확장을 설계하고 이를 구현을 통해 검증한다.

Keywords: 미래인터넷 테스트베드, 프로그래머블 네트워킹, OpenFlow, 네트워크 가시화 및 제어.

1. 서론

미래인터넷을 위한 중요한 기술 중 하나는 네트워크 가상화 기술이다[1]. 네트워크 가상화를 통해 연구자는 자신이 개발한 새로운 네트워크 기능과 알고리즘을 다른 연구자에게는 전혀 영향을 미치지 않으면서, 자유롭게 구현하고 실험할 수 있다. 이러한 네트워크 가상화를 구현하기 위한 한 가지 방법으로 프로그래머블 네트워크를 활용할 수 있다[2]. 스위치나 라우터 등 각종 네트워크 장치로부터 제어 소프트웨어를 분리해 내고, 개방형 프로그래머블 네트워크 인터페이스를 제공함으로써 새로운 서비스의 개발과 적용을 용이하게 지원한다.

최근 활발히 연구가 이루어지는 OpenFlow [3]는 송신된 트랙픽 중 특정한 플로우나 그들의 집합(flowspace)에 대해 연구자가 원하는 전송 경로를 직접 정의할 수 있는 기능을 제공하는 프로그래머블 네트워크를 지원한다. 예를 들면 스위치(라우터)에 들어오는 트래픽을 목적지에 따라 플로우의 형태로 구분하여 각각의 플로우에 대한 제어를 제공할 수 있다. 연구자는 OpenFlow 기반으로 구성된 테스트베드를 이용해 기존 네트워크의 정상적인 서비스를 간섭하지 않으면서 새로운 네트워크 기술을 연구하고 검증할 수 있다.

ENVI(Extensible Network Visualization & Control Framework)는 OpenFlow 기반 네트워크를 위한 가시화 및 제어 프레임워크이다[3]. 개발자가 특정 목적의 가시화 부분을 만들 수 있으며, 이를 통해 자신만의 (즉 가상화된) OpenFlow 네트워크의 토폴로지와 플로우들을 보면서 제어할 수 있다. 최근 국내에서도 OpenFlow 를 이용한 테스트베드 구성에 관한 각종 연구들이 진행 중이다. 하지만, OpenFlow 기반 테스트베드의 상태 정보를 제공하는 모니터링 도구의 확보가 아직 부족하다. 따라서 연구자가 실험의 진행 상황에 대한 정보를 확인하고 플로우 경로를 능동적으로 재설정하기 쉽지 않은 실정이다. OpenFlow 테스트베드 모니터링을 위한 Web 기반 GUI 를 제안했었지만[4], 테스트베드 토폴로지의 정보 표현이 OpenFlow 스위치에 한정되어 스위치에 연결된 호스트의 정보를 확인할 수 없고 기능 추가를 위한 프로그래밍의 확장성에 문제가 있었다[5]. 또한 OpenFlow 기반 테스트베드를 위한 ENVI 기반의 모니터링 툴의 구현도 실질적인 플로우의 가시화 및 경로 제어에 미치지 못했었다[6].

따라서 본 논문에서는 OpenFlow 기반 테스트베드의 정보들, 즉 OpenFlow 스위치, 호스트, 그리고 링크의

정보들을 보다 직관적으로 보여주는 모니터링 기능과 발생된 플로우를 가시화하고, 간단한 조작으로 플로우 경로를 재설정하는 제어기 응용의 개발에 주목한다. 즉 이를 위한 확장된 ENVI 기반의 실험자 도구를 구현하고, 이를 실제 테스트베드 환경에서 실험을 통해 검증한다. 이를 위해 본 논문에서는 먼저 2 절에서는 OpenFlow 기반 네트워크의 관련연구를 알아본다. 이어서 3 절에서 제안하는 응용 개발과 실험자 도구에 대한 구현에 대해 구체적으로 설명한다. 4 절에서 실험을 통해 정상적인 동작을 검증한 다음 5 절에서 본 논문을 정리한다.

2. OpenFlow 기반 네트워크

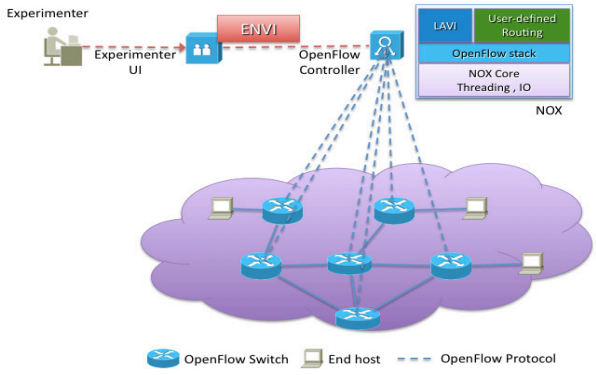

그림 1. ENVI 를 사용하는 OpenFlow 기반 네트워크.

그림 1 과 같이 일반적인 OpenFlow 기반 네트워크는 스위치와 라우터와 같은 역할을 하는 OpenFlow 스위치, 이를 자신이 원하는 목적에 맞게 명령, 제어할 수 있는 OpenFlow 제어기, 그리고 실험자에게 사용자 인터페이스를 제공하는 ENVI 로 구성된다.

2.1 OpenFlow

OpenFlow 는 각 플로우 구성요소를 연관지어 실행하는 Flow-table, OpenFlow 제어기로부터의 명령과 패킷을 보내고 받는 Secure Channel, 그리고 OpenFlow 제어기와 OpenFlow 스위치간의 통신을 지원하는 프로토콜로 구성된다[3]. OpenFlow 스위치는 OpenFlow 프로토콜에 준하여 OpenFlow 제어기와 연결을 형성/유지하면서 제어되는 하드웨어 스위치 장비이다. 상용 스위칭 장비들의 지원도 확산되고 있지만, 일반적으로는 다수의 네트워크 인터페이스를 지닌 PC 를 이용하여 구성한다. Secure Channel 은 OpenFlow 제어기와의 통신을 위한 TCP 기반의 secure 채널을 형성한다. Flow-table 은 VLAN, 이더넷, IP, TCP, 포트 정보들을 가지고 있으며, 이러한 정보들에 의해 플로우가 정의되고, Openflow 프로토콜에 근거해 정보들이 OpenFlow 제어기로 전달되면서 플로우 제어 동작이 결정된다.

2.2 NOX (Network OS)

OpenFlow 제어기는 OpenFlow 스위치들을 제어하는 역할을 담당하며, 주로 NOX 소프트웨어를 이용하여 일반 사양(specification)의 PC 장비로 구축할 수 있다. NOX 는 네트워크 컨트롤 플랫폼으로, OpenFlow 네트워크 상에서 네트워크 관리와 제어를 위한 응용을 구성하는 프로그래매틱 인터페이스를 제공한다[7]. NOX 는 NOX Core 와 NOX Components, 그리고 Events 로 구성된다. NOX Core 는 OpenFlow 스위치들과의 컨트롤 채널을 관리하고, OpenFlow 프로토콜을 이해하여 응용과 OpenFlow 스위치 사이에서 연결을 담당한다. NOX Components 는 재사용이 가능한 기능들에 대한 구현 요소들로 개발자는 이를 확장하고 조합하여 원하는 응용을 만든다. 또한 Events 는 OpenFlow 네트워크에서 발생하는 이벤트에 대한 정보를 NOX Components 와 연결시킨다.

2.3 ENVI

ENVI 는 OpenFlow 네트워크 가시화 및 제어를 위한 프레임워크이다[3]. OpenFlow 기반 네트워크를 가시화하고 특정 목적에 맞게 제어하는 사용자 인터페이스 개발 환경을 제공한다. ENVI 프레임워크는 세가지 주요 모듈로 구성된다. 첫째, OpeFlow 제어기 안에 위치한 서버의 정보를 가시화 부분으로 전달하고 사용자 상호작용의 정보를 서버에 전달하는 통신 모듈, 둘째 마우스의 클릭이나, 키보드의 입력과 같이 사용자의 명령을 표현하는 사용자 상호작용 모듈, 셋째 정보들을 최종적으로 보여주는 가시화 모듈이다.

3. 가시화 및 플로우 제어를 위한 NOX 응용 개발 및 ENVI 확장 구현

본 절에서는 OpenFlow 테스트베드의 가시화 및 플로우의 제어를 위해 NOX 플랫폼을 이용해 토폴로지 정보 수집과 경로 설정을 수행하는 응용의 개발과 사용자 가시화/제어 인터페이스를 제공하기 위한 ENVI 확장을 다룬다.

3.1 가시화 및 플로우 제어를 위한 NOX 응용 개발

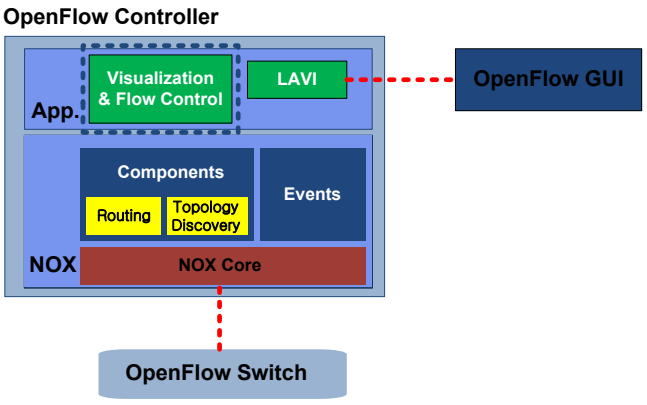

그림 2. 가시화 및 플로우 제어를 위한 제어기 구성.

그림 2 는 OpenFlow 테스트베드의 정보를 수집하여 구성된 네트워크를 가시화하고 발생된 플로우의 경로 제어를 위한 소프트웨어 및 하드웨어 구성도이다. NOX 플랫폼에 따른 응용을 포함한 OpenFlow 제어기

를 중심으로 OpenFlow 스위치들과 OpenFlow GUI 사용자 인터페이스로 구성된다.

가시화 및 플로우 제어를 위한 NOX 응용의 주요 기능은 세가지로 나뉜다. 첫째로 OpenFlow 테스트베드의 토폴로지 정보를 수집하고, 둘째로 플로우의 경로를 제어하며, 셋째로 OpenFlow GUI 와 통신을 하는 LAVI (Extensible Backend for Network Visualization)로 정보를 전달한다. 이를 위해 NOX 의 Topology Discovery 콤포넌트를 이용해 노드 정보와 노드들 간의 링크 정보를 수집하고, Routing 콤포넌트를 이용해 OpenFlow 스위치의 플로우 테이블에 명령을 전달한다. 각 Component 를 활용하려면 Event 호출이 필요하며, 호출이 적정한 시간에 연결되도록 개발한 응용에 적용시킨다. 연결을 위한 구현의 설계와 상세한 내용은 3.3 절에서 기술할 것이다.

3.2 가시화 및 플로우 제어를 위한 ENVI 확장

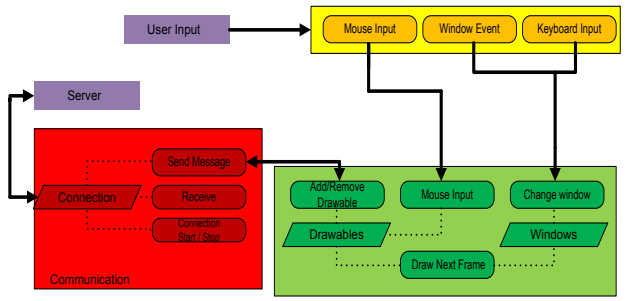

그림 3. ENVI 모듈의 상호작용 다이어그램.

그림 3 은 OpenFlow 제어기로부터 전달된 정보를 가시화하고 플로우 경로를 재설정하는 사용자 인터페이스를 제공하는 ENVI 기반의 OpenFlow GUI 구성도이다. 크게 세가지 모듈로 나누어 각 모듈을 통해 개발하는 NOX 응용을 위한 정보 가시화와 플로우 경로 제어를 용이하게 만들었다. 첫째, Communication 모듈은 서버(Server) 즉 OpenFlow 제어기의 LAVI 와 통신하는 Connection, 정보를 전달하는 Send Message, 연결을 제어하는 Connection stop/start 로 구성된다. 둘째, User Interaction 모듈은 마우스와 키보드의 입력을 각각 처리하는 Mouse Input 과 Keybord Input 으로 구성된다. 셋째, Display 모듈은 Communication 을 통해 전달된 OpenFlow 제어기의 토폴로지 및 플로우 정보들을 그래픽으로 추가/삭제하는 Add/Remove Drawable, 마우스 입력과 키보드의 입력을 받아 그래픽으로 표현하는 Mouse Input, 전체 그래픽를 위한 데이터 정보 저장을 위한 Drawables, 윈도우 창의 크기 변화를 조절하고 표현하는 Change window 와 Window 그리고 Drawables 의 데이터를 Windows 로 전달하는 Draw Next Frame 으로 구성된다.

3.3 가시화 및 플로우 제어를 위한 OpenFlow 스위치, 제어기 그리고 GUI 연결 인터페이스 설계

3.1 절과 3.2 절의 NOX 응용 개발과 ENVI 의 확장을 위해서 필요한 OpenFlow 스위치, OpenFlow 제어기, 그리고 OpenFlow GUI 간의 인터페이스들은 다음과 같다.

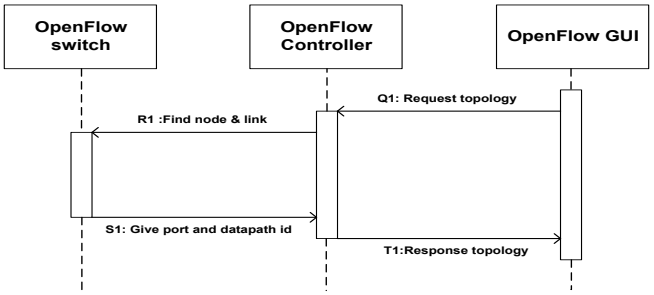

그림 4. 토폴로지 정보요청/응답 메시지 시퀀스 차트.

그림 4 는 토폴로지 가시화를 위한 메시징 절차를 보여준다. OpenFlow GUI 를 통해 토폴로지 정보를 요청하면, OpenFlow 제어기는 OpenFlow 스위치의 노드와 링크를 찾아 OpenFlow GUI 에게 OpeFlow 스위치의 아이디와 연결된 링크의 포트 정보를 알려준다.

표 1 은 토폴로지 정보 요청/응답을 위한 각 인터페이스의 ID, 이름, 그리고 데이터에 담고 있는 정보와 상세 내용을 설명한다.

<표 1> 토폴로지 정보 요청/응답 인터페이스.

ID	이름	데이터	설명
Q1	Request topology	Link & Node state specification	OpenFlow GUI 는 OpenFlow Controller 에게 topology 정보를 요청함.
R1	Find node & link	Link & Node state specification	OpenFlow Controller 는 OpenFlow Switch 의 노드와 링크를 검색함.
S1	Give port and datapath id	Link & Node state specification	OpenFlow Switch 는 OpenFlow Controller 에게 Datapath ID 와 포트 정보를 전달함.
T1	Respose topology	Link & Node state specification	OpenFlow controller 는 Datapath ID 와 포트 정보를 통해 토폴로지 정보를 수집하고 OpenFlow GUI 에게 토폴로지 정보를 전달함.

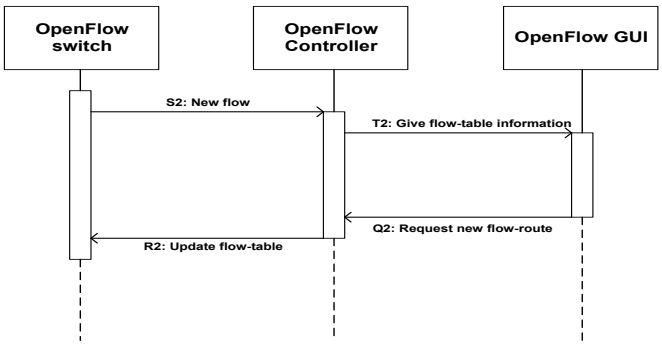

그림 5. 플로우 가시화 및 경로 재설정 메시지 시퀀스 차트.

그림 5 는 OpenFlow 테스트베드에서 발생된 플로우의 가시화 및 경로 재설정을 위한 메시징 절차이다. OpenFlow 스위치에서 플로우가 새로 발생하면 OpenFlow 제어기를 통해 플로우 테이블 정보를 수집하여 OpenFlow GUI 에게 전달을 통해 가시화한다. 새

로운 경로로 재설정이 요청되면 OpenFlow 제어기가 OpenFlow 스위치의 플로우 테이블 정보를 업데이트 하여 플로우의 경로를 갱신한다. 표 2 는 플로우 가시화 및 경로 재설정을 위해 인터페이스를 통해 전달되는 정보의 내용이다.

<표 2> 플로우 가시화 및 경로 재설정 인터페이스.

ID	이름	데이터	설명
S2	New flow	Flow specification	OpenFlow Switch 는 OpenFlow Controller 에게 새로운 flow 정보를 전달함.
T2	Give flow-table information	List of flow	OpenFlow Controller 는 OpenFlow GUI 에게 수집된 OpenFlow Switch 의 flow table 정보를 전달함.
Q2	Request new flow-route	List of flow	OpenFlow GUI 는 OpenFlow Controller 에게 flow 경로를 요청
R2	Update flow-table	Flow Data	OpenFlow controller 는 OpenFlow Switch 의 flow table 정보를 갱신함.

<표 3> 인터페이스에서 입출력되는 데이터들의 형식.

데이터 형식	내용
Flow specification	플로우를 명시하며, OpenFlow Switch 에 최초로 전송된 패킷헤더 정보를 통해 추출함.
List of flow	OpenFlow Switch 사이에 발생된 flow 의 정보 집합
Flow Data	플로우 테이블에 저장하기 위한 각 엔트리에 대한 정보 집합
Link state specification	OpenFlow Switch 와 물리적으로 연결된 링크의 상세 정보
Node state specification	OpenFlow Switch 의 내부 CPU/메모리 및 큐의 상세 정보

표 3 은 데이터 형식에서 사용되는 값들의 상세 설명을 나타낸다.

4. 구현 환경 및 결과

본 시스템의 구현 및 개발환경은 다음과 같다.

- 개발 언어 : C++, Python, JAVA
- 플랫폼 : OpenFlow, NOX
- 운영체제 : Cent OS 5.2

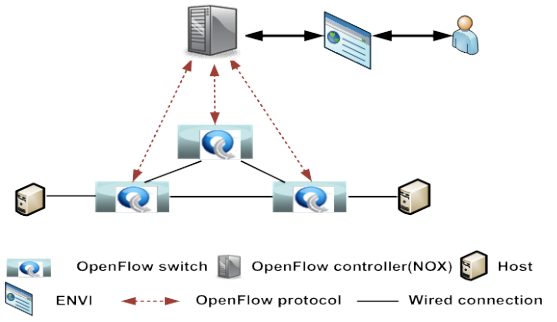

그림 6. 실험 테스트베드 구성도.

그림 6 은 구현 검증을 위한 실제 OpenFlow 테스트베드 구성도로써, OpenFlow 스위치 3 대, 호스트 2 대, NOX 및 ENVI 용 장치 1 대로 구성된다.

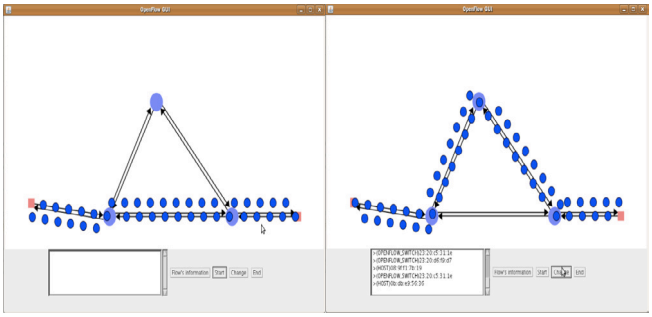

그림 7. ENVI 기반의 GUI: 플로우 변경 전/후

그림 7 은 OpenFlow GUI 의 모습이다. 실제 OpenFlow 테스트베드의 토폴로지 상태 및 플로우의 상태를 직관적으로 확인하고, 마우스와 버튼을 이용해 플로우의 경로를 쉽게 제어할 수 있다.

5. 결론

본 논문에서는 OpenFlow 테스트베드를 가시화하고 발생된 플로우를 쉽게 제어하도록 OpenFlow 제어기의 응용을 개발하고, ENVI 를 이에 따라 확장하였으며, 이를 실제 환경에서 검증하였다. 앞으로 실험자가 원하는 네트워크의 상태 정보나 OpenFlow 스위치 플로우 테이블 정보를 OpenFlow GUI 로 확인하고, 보다 다양한 OpenFlow 테스트베드를 쉽게 제어할 수 있도록 개선하는 것이 필요하다.

감사의 글

본 논문은 “미래인터넷 인프라를 위한 가상화 지원 프로그래머블 플랫폼 및 핵심원천 기술개발” 과제(2009-F-050-01)에 대한 결과물 중 일부분으로 연구개발 업무에 도움을 주신 모든 분들께 감사 드립니다.

참 고 문 헌

[1] N. Chowdhury and R. Boutaba, *A survey of network virtualization*, University of Waterloo, Technical Report CS-2008-25, Oct. 2008.

[2] A. T. Campbell et. al., “A survey of programmable networks,” *ACM Computer Communications Review*, vol. 29, no. 12, pp. 7-23, Apr. 1999.

[3] ENVI: An Extensible Network Visualization & Control Framework, http://www.openflowswitch.org.

[4] N. McKeown, et. al., “OpenFlow: Enabling innovation in campus networks”, *SIGCOMM Computer Communications Review*, vol. 38, Issue2, Mar. 2008.

[5] 신성호, 김남곤, 김종원, “OpenFlow 스위치 노드 모니터링을 위한 Web 기반 GUI 의 구현,” 신호처리합동학술대회 논문지, vol. 22, pp. 27-30, Oct. 2009.

[6] 신성호, 김남곤, 김종원, “OpenFlow 테스트베드를 위한 ENVI 기반 모니터링 툴 구현,” 정보통신합동학술대회, vol. 3, pp. 267-271, Nov. 2009.

[7] N. Gude et. al., “NOX: Towards an operating system for networks,” *ACM SIGCOMM Computer Communications Review*, vol. 38, no. 3, July 2008.