

미래 인터넷 테스트베드에 대한 연구

권윤주^o

한국과학기술정보연구원
yulli@kisti.re.kr

Study on the Future Internet Testbed

Yoonjoo Kwon^o

Korea Institute of Science Technology Information(KISTI)

요 약

현 네트워크의 한계를 극복하고, 미래사회의 기반으로서의 역할을 할 수 있는 네트워크인 미래 네트워크에 대한 연구가 최근 활발히 진행되고 있다. 이러한 미래 네트워크에 대한 시도들이 이론으로만이 아니라 실제 네트워크상에서 구축되기 위해서는 미래 네트워크를 위한 테스트베드가 필요하다. 특히 미래 네트워크 테스트베드는 현재 네트워크와 공존뿐만이 아니라, 다양한 미래 네트워크 시도들을 적용시킬 수 있는 형태로 구성되어야 의미가 있다. 하나의 공유인프라에 다수 개의 독립적인 네트워크를 구성해 주는 기술이 바로 네트워크 가상화이며, 이는 미래네트워크 실험 인프라를 구현하기 위한 핵심 기술 중의 하나이다. 본 논문에서는 네트워크 가상화 기술의 동향을 분석해보고 미래인터넷 테스트베드에 대하여 고민해보고자 한다.

1. 서 론

최근 '가상화'라는 개념이 여러 분야에서 대두되고 있다. 가상화는 응용 프로그램 또는 사용자에게 자원의 물리적 특성을 보이지 않게 하고 논리적 자원만 보이게 하는 기술로 정의할 수 있는데 이는 서버 가상화, 스토리지 가상화 등의 분야에서 시작하여 현재는 네트워크 가상화라는 개념까지 확대되고 있다[6].

특히 네트워크 가상화 기술은 애플리케이션과 네트워크 사이에 가상화 계층을 도입하여 애플리케이션과 네트워크 내 특정 자원간 밀접한 관계를 끊음으로써 하나의 네트워크 인프라상에서 다양한 요구사항을 갖는 응용별 네트워크들이 공존할 수 있는 기술이다. 네트워크 가상화는 다양한 네트워크 아키텍처 연구를 시험할 수 있는 테스트베드 구축 시 필요한 주요한 기술이다. 현재 이러한 네트워크 가상화 기술은 미래 인터넷을 염두에 두고 글로벌 테스트베드를 구축하고 있는 GENI(Global Environment for Network Innovations)의 기반으로 사용되고 있다.

본 논문에서는 기존의 네트워크 가상화 기술들을 분석하면서 국내 연구망특히 KREONET에서 구축해야 할 미래인터넷을 위한 테스트베드를 고민해보고자 한다

2. 네트워크 가상화 기술

GENI는 이전의 PlanetLab과 VINI의 성공적 연구경험을 기반으로 추진되고 있다. PlanetLab은 네트워크 3계층 이상에서의 새로운 기술의 테스트를 제공하고 있는 글로벌 연구 네트워크로서 2500명 이상의 사용자와 600개 이상의 슬라이스에 의하여 가상화 또는 자원할당 등의 테스트를 수행해온 경험을 제공한다. VINI는 PlanetLab의 경험을 2계층으로 옮겨서 도전하는 것으로 현재 National LambdaRail, Internet2에 연결되어 있는 26개 사이트에 42개 노드로 구성되어 있다. 이에 본 절

에서는 미래 네트워크 테스트베드의 원천기술이 되는 네트워크 가상화 기술인 PlanetLab과 VINI, Trellis에 대하여 기술하고자 한다.

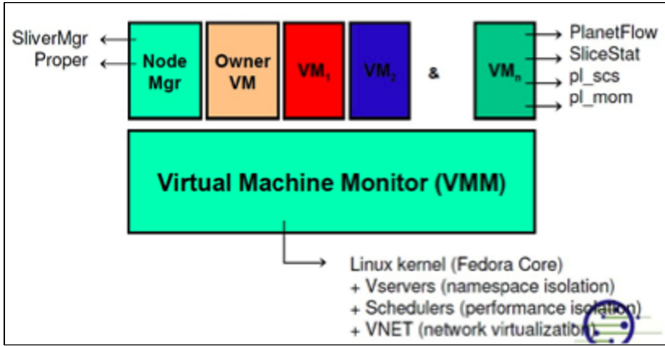
2.1 PlanetLab

PlanetLab은 전세계 연구자 커뮤니티에 의해 공유되는 전세계 규모의 분산 컴퓨팅 플랫폼이다. 이것은 구현 기반 연구 검증 테스트베드로서 기존의 많은 연구들이 시뮬레이터, 에뮬레이터 혹은 소규모의 테스트베드에 기반하여 검증이 이루어진 데 반해 글로벌 규모의 실험환경을 연구자들에게 제공할 수 있다. PlanetLab은 2003년에 사용자한테 컴퓨터를 기부받는 형식으로 구축하여 현재 473개 사이트에 953개의 노드가 연결되어 있다.

이러한 PlanetLab의 목표는 첫째, 연구자들이 네트워크 서비스를 세계적인 규모로 실험할 수 있는 플랫폼을 제공하는 것이며, 둘째 새로운 네트워크 서비스가 실제 사용자 커뮤니티에 적용되고 서비스될 수 있는 플랫폼을 제공하는 것이고, 나아가 서비스 기반 구조로 인터넷의 진화를 이루어 나가고자 하는 것이다.

2.1.1 PlanetLab의 구조 및 동작원리

PlanetLab은 개인 또는 기관이 기부한 컴퓨터들을 위에서 사용자들마다 독립된 연구 검증 환경을 만들어주는 분산 오버레이 네트워크 테스트베드이다. 각각의 기부된 컴퓨터는 하나의 노드(Node)로 관리되고, 네트워크 서비스 연구를 하고자하는 개인은 자신이 필요로 하는 노드들에 일정자원을 할당받아 '슬라이스(Slice)'라고 하는 개인 연구 영역을 확보할 수 있다.

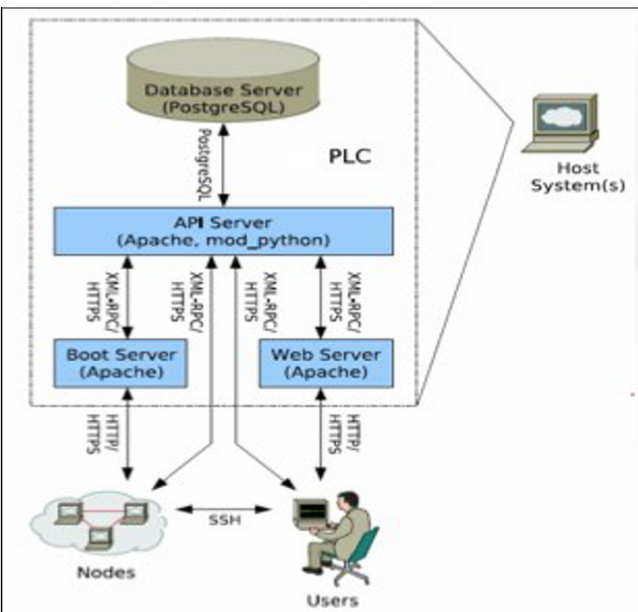


[그림 1] PlanetLab 노드 구조[1]

PlanetLab 노드는 [그림 1]에서 보는 바와 같이 구성되어 있다. 구성 요소로 노드관리자(Node Manager), OwnerVM, VMM(Virtual Machine Monitor) 그리고 다수 개의 VM(Virtual Machine)들이 있다[1]. 실질적으로 사용자에게 서비스되는 요소는 VM이고 Node Manager, OwnerVM은 전체적으로 PlanetLab 서비스를 운영하기 위한 노드 내 관리요소들이다

마지막으로 VMM은 한 노드 상에 다수의 VM을 만들어주는 노드 가상화를 수행하는 역할을 하는 요소이다 VMM은 Linux Kernel을 기반으로 하여 Vserver, Scheduler 그리고 VNET를 통해 레이어별 가상화를 수행한다. 우선 Linux Vserver는 OS레벨의 가상화를 담당하는 데, file시스템이나 CPU같은 OS자원을 파티셔닝하여 VM간 독립성을 지원하도록 한다 그리고 VNET은 네트워크 가상화를 담당하고 있는데 리눅스의 netfilter 시스템을 이용하여 구현되었으며 슬라이스와 패킷의 bound 및 connection tracking을 통해 패킷들이 해당하는 슬라이스에서 전송수신되도록 한다. 그리고 전송 네트워크 밴드위스에 대하여 "fair share"를 보장한다.

PlanetLab은 [그림 1]과 같은 구조를 갖고 있는 전세계에 퍼져있는 노드들과 하나의 제어노드인 PLC(PlanetLab Consortium/Central)로 구성되어 있다.



[그림 2] PlanetLab 서비스 구조(PLC 역할)

사용자(User)는 이러한 PLC를 통하여 PlanetLab 노드를 제공한다거나 슬라이스를 할당받는다거나하는 서비스를 제공할 수 있다 PLC 안에는 Boot 서버, 웹서버, 데이터베이스 관련 서버가 존재하여 PlanetLab노드를 관리하고, 사용자가 PlanetLab 데이터베이스에 등록되어 있는 노드들을 선택하여 사용할 수 있도록 서비스하고 있다. 따라서 실질적으로 사용자(User)는 PLC를 통해 자신의 슬라이스를 생성하고 생성된 슬라이스에 PlanetLab 노드들을 추가함으로써 자신의 슬라이스로 접속하여 사용할 노드의 집합을 구성하여 PlanetLab 테스트베드를 이용할 수 있다

2.2 VINI(Virtual Network Infrastructure)

기존의 네트워킹 연구 검증 인프라로 이용되고 있는 NS2나 emulab등은 시뮬레이션 또는 에뮬레이션 환경으로서 연구자 임의의 네트워크 구성에 인위적인 이벤트 발생을 통해 실험 결과를 도출하는 형태의 검증이었다 이에 PlanetLab개발 그룹의 일부진영에서는 VINI라는 네트워킹 연구 검증을 위한 인프라를 구축하였다 이것은 네트워킹 연구 검증을 하는 데 있어 시뮬레이션 환경의 장점인 "control"과 실제 테스트베드의 장점인 "realism"을 제공하고자 설계되었다[2]. 따라서 VINI는 분산된 서비스들로부터 발생하는 실제 트래픽이 존재하는 실제 네트워크와 같은 환경에서 다수의 새로운 프로토콜과 서비스들을 동시에 검증해볼 수 있도록 공유 인프라를 제공하고자 한다

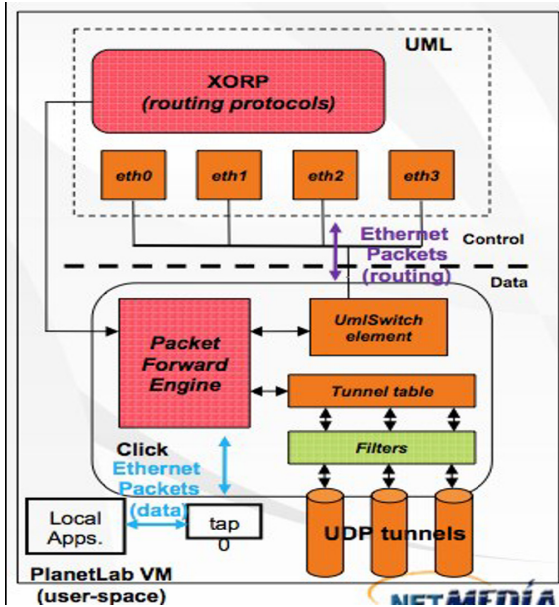
2.2.1 VINI 설계 요구사항

- 연구자 임의의 네트워크 토폴로지 생성
- 생성된 토폴로지 상에서의 패킷 전송을 위한 포워딩, 라우팅 기능 지원
- 실제 환경 구축을 위한 실제 외부 호스트에 대한 연결 지원
- 여러 실험의 동시 진행 지원

2.2.2 PL-VINI : PlanetLab Extensions for VINI

VINI는 2.2.1의 요구사항을 갖고 PlanetLab에서 VINI의 초기 프로토타입을 만들었다 PL-VINI는 VINI를 쉽게 프로토타이핑하기 위해서 PlanetLab OS에 대한 변화는 제한하면서 사용자영역에서 라우팅 및 포워딩에 관련된 대다수의 중요 기능들을 구현하였다 또한 VINI에서 제공하고자 하는 네트워크 실험들을 지원하기 위해서 다음과 같이 PlanetLab의 일부 기능들을 확장하기도 하였다.

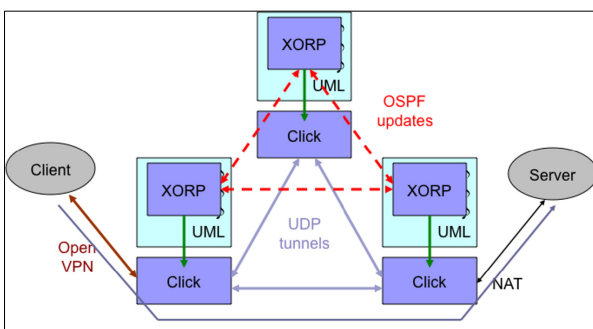
특히 PL-VINI는 [그림 3]에서 보는 바와 같이 PlanetLab VM이 위치하고 있는 사용자공간(user-space)에서 가상 네트워크를 구성하는 기법을 도입했다. 이는 UML(User-Mode Linux)를 이용하여 사용자공간(user-space)상에 가상의 네트워크 디바이스를 생성하고, 리눅스의 TUN/TAP 드라이버를 수정하여 로컬 응용이 구성된 오버레이 상에서 패킷을 송수신을 하는 방식으로 구현되었다. 이를 통해 PL-VINI는 VM간 네트워크 독립성을 제공하고 있다



[그림 3] PL-VINI (노드) 구조[3]

[그림 4]는 2.2.1에서 언급한 요구사항을 만족시키기 위해서 PL-VINI의 “IIAS(Internet In A Slice)”라고 하는 하나의 서비스 구조를 보여준다 IIAS는 5가지 컴포넌트로 구성되어 있다.

- IIAS의 패킷 포워딩 엔진(an overlay router) : Click
- IIAS의 라우팅 메커니즘(a control plane) : XORP
- IIAS로의 패킷 전송 방식(an overlay ingress) : OpenVPN
- IIAS로부터 패킷 수신 방식 (an overlay egress) : NAT
- IIAS를 구성하는 분산되어있는 시스템의 집합



[그림 4] PL-VINI 서비스 방식(IIAS)

[그림 4]와 같이 각각의 노드(VM)는 패킷 포워딩 엔진(Click)과 라우팅 메커니즘(XORP)를 가지고 하나의 라우터와 같이 기능한다 그리고 VM들은 UDP 터널을 이용하여 VM 간 가상 링크를 구성하고 각각의 VM에서 동작하고 있는 XORP를 이용하여 네트워크를 구성하며 지속적으로 라우팅 테이블을 갱신하고 Click을 이용하여 패킷 포워딩을 수행하게 된다 그리고 IIAS망에 접속하기 위하여 Client는 OpenVPN을 이용하고 IIAS 망에서 외부로 패킷이 전달되기는 위해서 NAT기법을 이용한다

2.3 Trellis

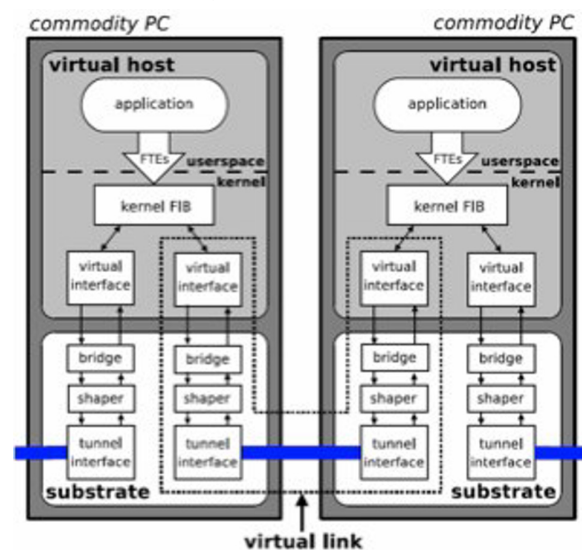
Trellis는 PL-VINI의 네트워크 가상화 성능을 향상시키는 한 방법으로 제안되었다 PL-VINI에서의 패킷 포워딩은 사용자 공간(user-space)에서 이루어졌기 때문에 오버레이 네트워크 기법을 이용하게 되는 데오버레이는 고속으로 트래픽을 전달하기에는 여러 가지 제한이 뒤따른다. 이에 프린스턴대학조지아텍/TU Berlin의 연구진들은 ‘Trellis’ 라는 공유되는 물리 네트워크 구조상에서 프로그래밍 가능한 다수개의 가상 네트워크를 운영할 수 있는 network hosting platform을 제안하게 되었다. 이것은 다음과 같이 5가지 특성을 목표로 한다

- Speed : 수 Gbps정도로 패킷을 전송할 수 있는 가상 네트워크 성능
- Isolation : 네트워크 자원과 시스템 자원의 네임스페이스 및 자원 분리 지원
- Flexibility : 라우팅 프로토콜을 포함한 네트워크 서비스에 대한 강력하고 친숙한 개발환경 제공
- Scalability : 개발 및 관리 비용을 줄일 수 있도록 동시에 다수의 가상 네트워크(virtual network) 지원
- Low cost : 가상 네트워크 호스팅 비용을 줄여 진입장벽을 낮추기 위해, 기존의 시스템에 가상 네트워크 호스팅

2.3.1 Trellis 구조

Trellis는 general-purpose OS를 사용하는 일반 하드웨어(commodity hardware)에서 운영될 수 있는 가상 네트워크 장치(substrate)이다[4]. 이것은 앞서 언급했던 ‘VINI 설계 요구사항’ 중 사용자 공간(user-space)에서의 패킷 포워딩 및 라우팅에 대하여 커널 기반으로 구현함으로써, PL-VINI의 성능을 많이 개선하였다

- Trellis 노드 구조



[그림 5] Trellis의 구조

Trellis는 [그림 5]와 같이 노드를 구조하였다 우선 각 VM은 가상호스트 지원 계층과 substrate 계층으로

구성되고, 가상호스트 계층은 사용자 공간(user-space)과 커널 공간을 포함하여 가상 네트워크의 라우팅 및 포워딩 계층을 커널 기반으로 가능하게 하였고 substrate 계층은 기존 물리 영역에서 지원되었던 호스트간 연결을 가상화시켜 가상호스트간 연결(가상 링크)을 지원하도록 설계되었다. 가상 호스트 내의 엘리먼트들은 가상 호스트 응용에 의해서 접근 및 제어될 수 있지만 substrate 내의 엘리먼트들은 직접 접근할 수 없고 Trellis Management 소프트웨어에 의해서만 설정될 수 있다

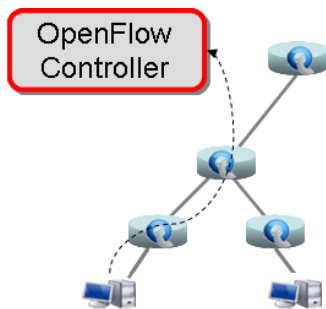
3. 미래 인터넷 테스트베드

전세계적으로 미래 인터넷을 지원하는 테스트베드는 하나의 물리적 네트워크 자원을 네트워크 가상화 기술을 이용하여 다수의 논리적 네트워크로 사용하는 것을 기본으로 하고 있다. 그 이유는 미래 인터넷을 위해 다양하게 시도되는 접근들을 실제 망에서 적용시켜봄으로써 실제로 그 기술이 인터넷에 도입되었을 때 문제 또는 성능들을 실제에 가깝게 분석하기 위함이다

다수의 시도들을 실제 네트워크에 적용하기 위해서는 일차적으로 실제 트래픽과 실험 트래픽으로 나누어 서로 독립적으로 라우팅할 수 있는 기술이 필요하고 또 더 다양한 미래인터넷 시도들을 적용시키기 위해서 실험 트래픽들도 독립적으로 라우팅할 수 있는 기술들이 필요하다.

앞서 언급한 네트워크 가상화 기술들은 네트워크 코어에서 적용할 수 있는 기술들은 아니고 종단 호스트들에서 중간의 네트워크 코어(백본)을 배제하고 네트워크 실험환경을 만들 때 사용할 수 있는 기술들이다 그러나 현재의 연구되고 있는 미래인터넷의 다양한 시도들을 수용하기 위해서는 네트워크 코어(백본)자체도 가상화 할 수 있는 기술이 필요하다. 본 논문에서는 OpenFlow를 이용한 미래인터넷 테스트베드 구축을 제안하고자 한다

3.1 제안하고자하는 미래 인터넷 테스트베드



[그림 6] OpenFlow를 이용한 미래 인터넷 테스트베드

현재의 인터넷은 'IP 주소'를 기본으로 송신지부터 목적지까지 패킷을 배송하는 '라우팅'을 수행한다. 따라서 미래 인터넷에서는 현재 인터넷과 같은 체계로 IP 주소를 기본으로 하여 패킷을 전송할 수도 있지만 또 다른 라우팅 메트릭을 기준으로 라우팅을 수행할 수 있다. 이러한 다양한 라우팅 방식을 수용하기 위해서는

OpenFlow 기술을 이용한 미래 인터넷 테스트베드를 제안하고자 한다.

[그림 6]과 같이 OpenFlow(네트워크 테스트베드는 다수의 switch와 하나의 controller로 구성되어 있으며, switch는 포워딩하는 역할을, controller는 해당 네트워크에 진입하는 패킷들에 대한 라우팅을 결정하는 역할을 한다.

기존 라우터가 'IP주소'를 가지고 라우팅을 결정하는 반면, OpenFlow의 controller는 [그림 7]과 같이 10개의 튜플을 이용하여 라우팅을 결정한다. 이때, controller는 10개의 튜플을 모두 사용할 수도 있고 자기가 사용하지 않는 요소에 대해서는 'wildcard'를 이용하여 배제시킬 수도 있다.

| Ingress port | Ether src | Ether dst | Ether type | VLAN id | IP src | IP dst | IP proto | TCP/UDP src port | TCP/UDP dst port |
|--------------|-----------|-----------|------------|---------|--------|--------|----------|------------------|------------------|
|--------------|-----------|-----------|------------|---------|--------|--------|----------|------------------|------------------|

[그림 7] OpenFlow의 flow entry header[5]

OpenFlow 네트워크는 포워딩과 라우팅 결정 역할이 물리적으로 분리되어 있기 때문에 새로운 라우팅 기술을 적용시키기 위해서 패킷 전달노드인 스위치와 상관없이 라우팅 결정역할을 하는 controller의 구현만을 변경시키면 된다. 따라서 하나의 네트워크 인프라에서 다양한 미래인터넷 기술들을 시도하는 테스트베드로서 OpenFlow 네트워크 테스트베드가 유용한 기반이 된다

4. 결론

현재 우리가 사용하고 있는 인터넷은 확장성 문제 보안 문제 등을 이유로 해서 새로운 인터넷이 등장할 때가 되었다고 얘기하고 있다. 이러한 이유로 세계적으로 '미래 인터넷'이라는 키워드로 많은 연구들을 하고 있지만 실제로 이 연구들을 실제 네트워크에 적용시키는 것도 하나의 큰 이슈로 제기되고 있다

이에 따라 미국에서는 GENI, 유럽에서는 FP7, 일본에서는 AKARI라는 이름을 가지고 미래인터넷을 수용할 수 있는 준비를 하고 있지만 아직 우리나라에서는 미래 인터넷에 대한 연구외에 그 기술을 담은 테스트베드에 대한 고민은 부족한 현실이다. 이에 본 논문에서는 미래 인터넷 테스트베드를 위한 기술로서 네트워크 가상화 기술들을 살펴보고, 네트워크 코어에서 적용할 수 있는 미래 인터넷 테스트베드를 제안하였다. 향후 계획으로는 KREONET 일부 구간에 OpenFlow 네트워크를 이용한 미래 인터넷 테스트베드를 구축하고자 한다.

[참고문헌]

[1] Larry Peterson and Timothy Roscoe, "The Design Principles of PlanetLab", Operating Systems Review, 40(1):11~16, January 2006
 [2] Andy Bavier, Nick Feamster, Mark Huan, Larry Peterson, Jennifer Rexford, "In VINI Veritas : Realistic and Controlled Network Experimentation", Proc. ACM SIGCOMM, August 2006

- [3] 이현용, 안서현, 김중원, “VINI(PlanetLab 가상화 확장) 이행 및 활용 방안” , Future Internet Winter Camp 2009
- [4] Sapan Bhatia, Murtaza Motiwala, Wolfgang Muhlbauer, et al., “Trellis: A platform for building flexible, fast virtual networks on commodity hardware“, in Proc. Workshop on Real Overlays and Distributed Systems(ROADS), December 2008
- [5] Brandon Heller, “OpenFlow switch Specification version 0.8.9“, December 2008
- [6] 권순중, “미래인터넷을 위한 네트워크 가상화 기술” , 정보과학회지, 제26권, 제10호, pp.76~81, 2008