

VINI (PlanetLab 가상화 확장) 이해 및 활용 방안

HyunYong Lee*, SuHyun Ahn, and Dr. JongWon Kim
Networked Media Lab., Dept. of Information and Communications
GIST (Gwangju Institute of Science & Technology), Korea
hrlee@nm.gist.ac.kr

2009. 2. 24



Contents

PlanetLab 가상화 확장

- PlanetLab
- VINI - PL-VINI
- VINI - Trellis

Private VINI (MyVINI)

- 설치 및 검증
- 운용 및 확장



PLANETTLAB 가상화 확장

PlanetLab의 구조, 가상화 및 제약점

Private PlanetLab 테스트베드들의 등장

VINI: PL-VINI, VINI-Trellis



PlanetLab (1/2)

PlanetLab (PL)의 이해

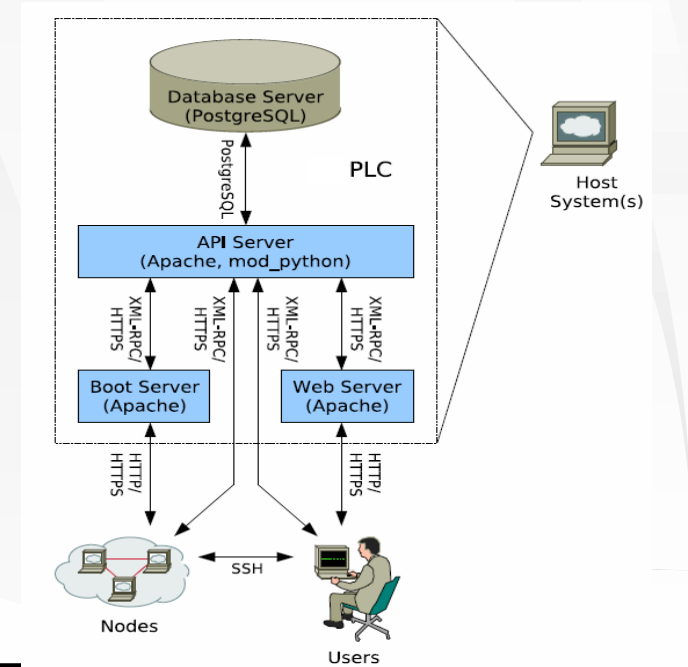
- 구현 기반 연구 검증 테스트베드: "PlanetLab is a global research network that supports the development of new network services"
 - ☞ 기존의 많은 연구들이 시뮬레이터, 에뮬레이터, 혹은 소규모의 테스트베드에 기반하여 검증이 이루어짐 → 글로벌 규모의 PlanetLab 테스트베드를 활용함으로써, 대규모의 환경에서 구현 기반의 연구 검증을 할 수 있게 되었다. 또한 가상화 기술을 통해 다수의 연구자들이 동시에 다발적으로 자신들의 연구를 진행할 수 있다.
- 2003년에 구축 시작. 현재는 473 사이트의 953개의 노드가 연결되어 있음
- PlanetLab 노드용 머신 2대 이상 지원함으로써 PlanetLab 테스트베드에 참여할 수 있음 (Non-profit 기관의 경우)

- 구조: PLC (DB server + API server + Boot server + Web server)
+ PlanetLab 노드들 + PlanetLab 사용자

역할 구분

- ☞ **PI (Principal Investigator)**: Slice / 계정 / 노드 관리
- ☞ **Technical Contact**: 실제적인 노드 관리, 문제 해결 담당
- ☞ **User**: PI, Technical Contact의 도움을 받아 PlanetLab 노드를 사용하는 사용자

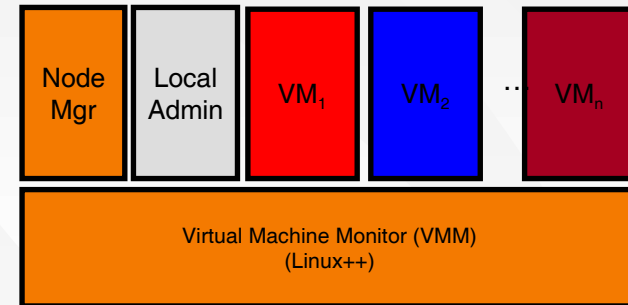
- PlanetLab 노드 사용: 사용자는 자신의 slice를 생성하고, 생성된 slice에 PL 노드를 추가(association) 함으로써 자신의 slice로 접속하여 사용할 노드의 집합을 구성한다.



PlanetLab (2/2)

PlanetLab의 가상화

- 컴퓨팅 가상화
 - ☞ 리눅스 VServer를 사용함.
 - ☞ "fair share" of the node's available CPU & temporary share increases (via Sirius)
- 네트워킹 가상화
 - ☞ 리눅스 Netflier 시스템을 활용하는 VNET을 구현
 - ☞ Slice와 패킷의 bound 및 connection tracking을 통해 패킷들이 해당하는 slice에서 전송/수신되도록 한다.
 - ☞ "fair share" access to outgoing network bandwidth



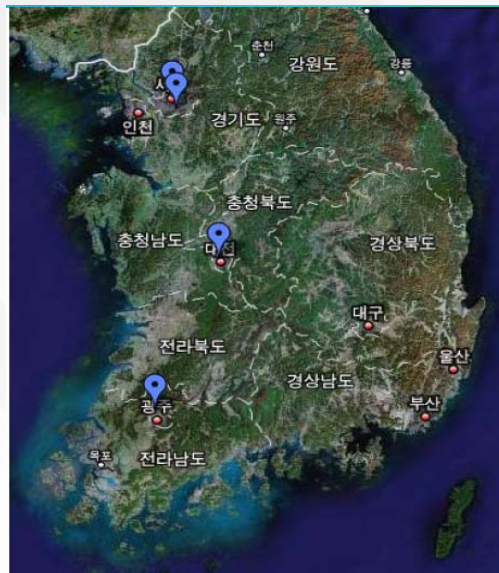
PlanetLab의 제약점

- 연구자가 원하는 기능을 임의로 추가하기 어렵다.
- 연구자가 원하는 임의의 네트워크 환경을 꾸미기 어렵다.
- 가상화를 통한 성능 보장을 기대하기 어렵다.
-

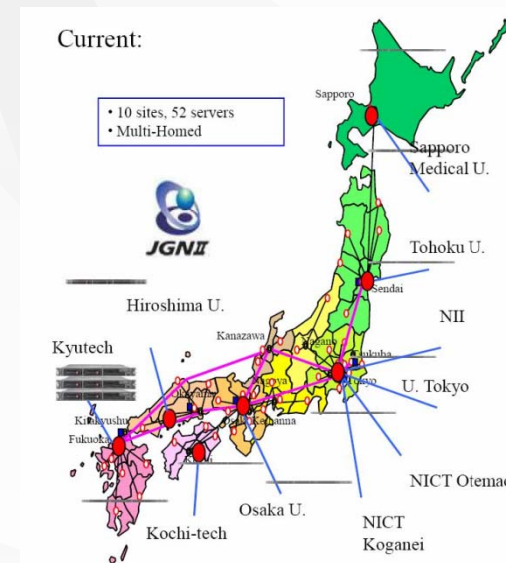
Private PlanetLab

연구자가 자신만의 목적을 위해 개발한 PLC (MyPLC)에 기반하여 구축된 PlanetLab 테스트베드

- Intel's Wireless PlanetLab – PlanetLab for wireless research (France)
- CoreLab – PlanetLab-based more flexible environment supporting enhanced resource virtualization
- EverLab – Private PlanetLab for HPC and Networking researchers (EU & Israel)
- VINI – PlanetLab for layer 3 researches, enhanced computing/networking virtualization
- PPK (Private PlanetLab Korea)



PPK (Korea)



CoreLab (Japan)

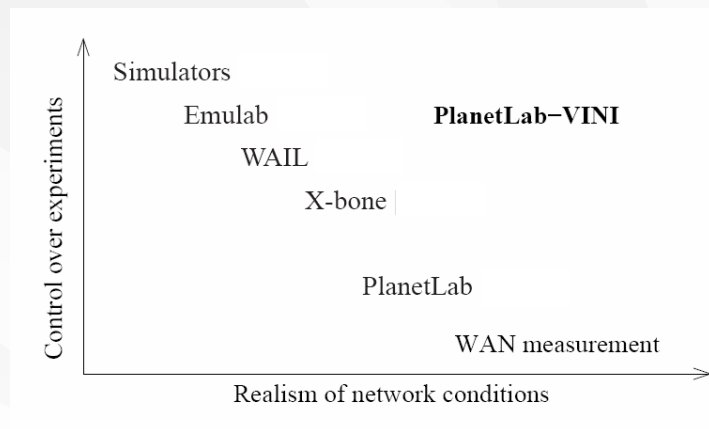
VINI (VirtuaL Network Infrastructure) – PL-VINI (1/3)

VINI (PL-VINI)의 배경

- 시뮬레이션 환경의 장점인 "control" + real testbed의 장점인 "realism"
- New techniques to support improved virtualization

VINI (PL-VINI)의 가상화

- 컴퓨팅 가상화: Improved CPU isolation
 - ☞ VINI는 PlanetLab의 "CPU reservation"와 "Linux real-time priorities"를 활용하여 보다 향상된 CPU isolation을 제공한다.
- 네트워킹 가상화: Virtual network devices
 - ☞ 각 user-space에 가상의 네트워크 디바이스를 생성하기 위해 UML (User-Mode Linux)를 사용
 - ☞ 리눅스 TUN/TAP 드라이버를 수정하여, 로컬 응용이 구성된 오버레이 상에서 패킷을 송/수신 하도록 하였다.



VINI (Virtual Network Infrastructure) – PL-VINI (2/3)

VINI (PL-VINI)의 이해

■ 설명

☞ “a virtual network infrastructure that allows network researchers to evaluate their protocols and services *in a realistic environment* that also provides *a high degree of control* over network conditions”

■ 지원 기능

☞ 연구자 임의의 네트워크 토폴로지를 생성 가능

- Virtual network devices in UML
- UDP tunnels and encapsulation in Click

☞ 생성된 토폴로지 상에서의 패킷 전송을 위한 포워딩, 라우팅 기능 지원

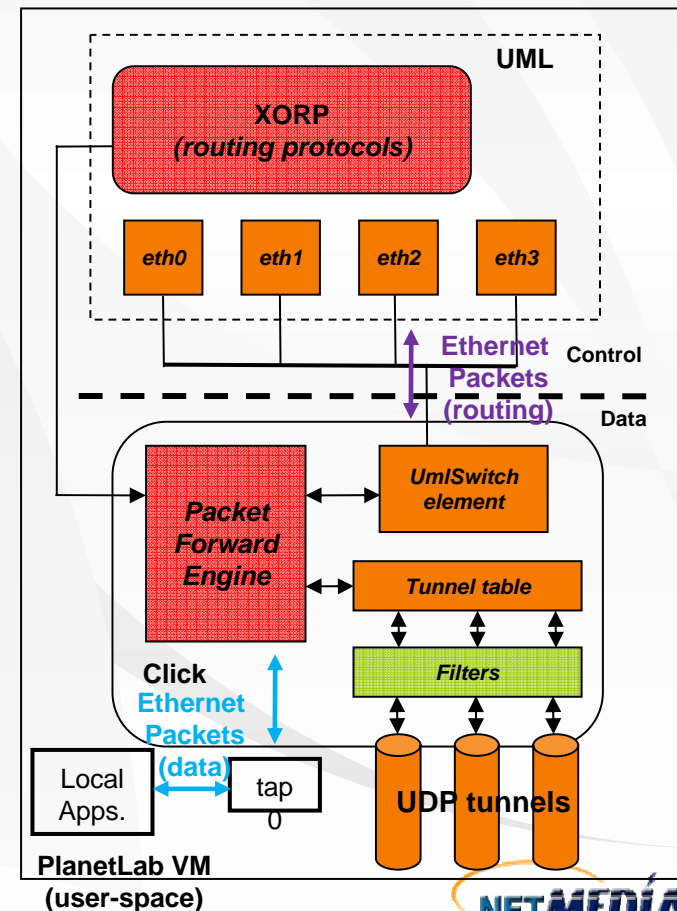
- Separate instance of Click/XORP on each virtual node
- Click for forwarding (data plane)
- XORP for routing (control plane)

☞ Realistic condition 구축을 위한 external real end hosts에 대한 연결 지원

- End-host connection to an OpenVPN server
- NAT in Click on egress

☞ 여러 실험의 동시 진행 지원

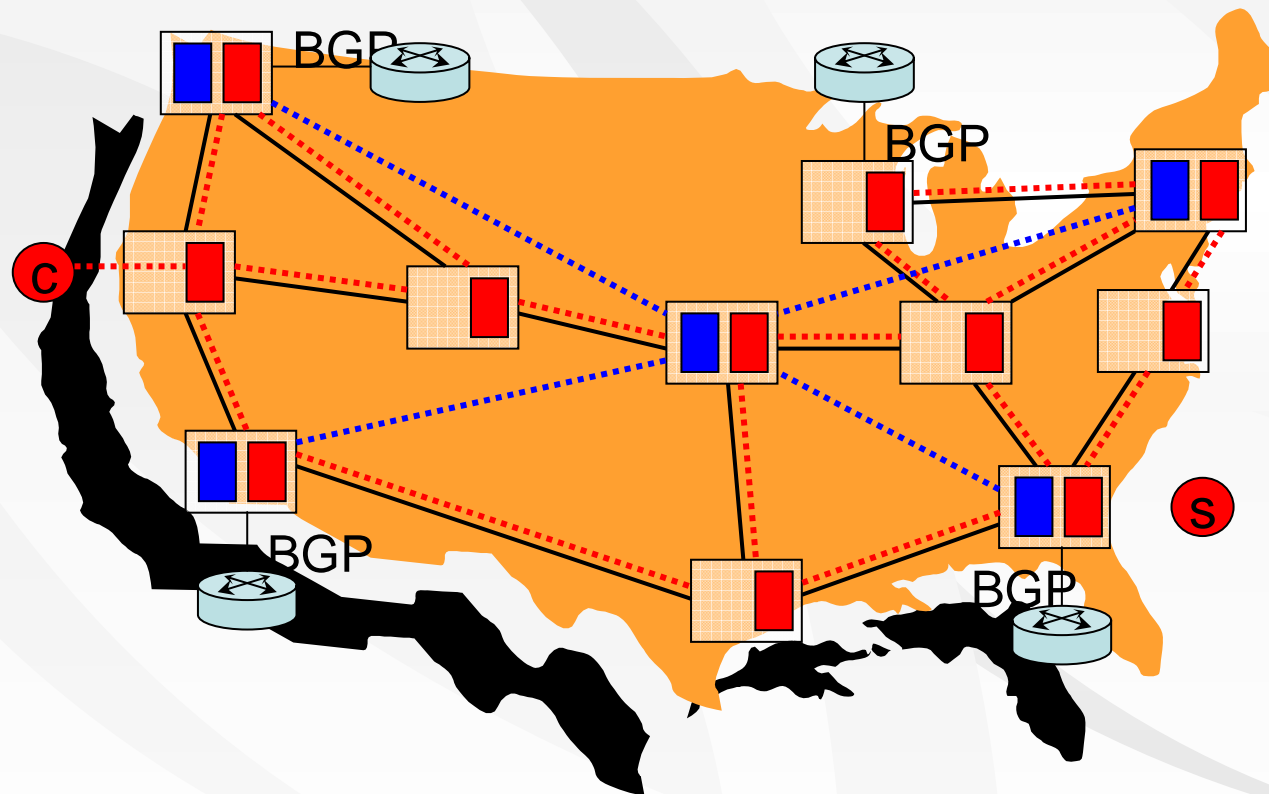
- Virtual servers and network isolation in PlanetLab
- Extensions for CPU reservations and priorities



VINI (Virtual Network Infrastructure) – PL-VINI (3/3)

VINI (PL-VINI)의 이해

- “arbitrary network topologies on a shared physical infrastructure” + “carrying real traffic” + “controlling network events”



VINI (VirtuaL Network Infrastructure) - Trellis (1/2)

- “Allows each virtual network to define **its own topology, control protocols, and forwarding tables**, which lowers the barrier for deploying custom service on an isolated, reconfigurable, and programmable network”
- Design goals: Speed / Isolation / Flexibility / Scalability / Low Cost
- Trellis의 배경:
 - PL-VINI의 네트워크 가상화 성능을 향상시키는 한 방법으로 제안됨
 - ☞ VINI에서의 패킷 포워딩은 user space에서 이루어졌기 때문에 좋은 성능을 내지 못하였다. Trellis에서는 패킷 포워딩이 kernel space에서 이루어지도록 함으로써 성능 향상을 이루고자 하였다.
- Trellis의 가상화
 - 컴퓨팅 가상화
 - ☞ PL-VINI와 동일
 - 네트워킹 가상화
 - ☞ Kernel space에서의 패킷 포워딩을 지원하기 위해 NetNS를 사용
 - ☞ User-space의 응용으로 하여금 kernel의 FIB를 직접 수정할 수 있도록 하였음
 - ☞ 고속의 가상 네트워크 지원을 위해 새로운 터널링 기술인 EGRE를 GRE에 기반하여 개발함
 - ☞ Point-to-point 연결의 성능 향상을 위해서 shortbridge를 개발



VINI (Virtual Network Infrastructure) - Trellis (2/2)

Trellis의 네트워크 가상화

EGRE

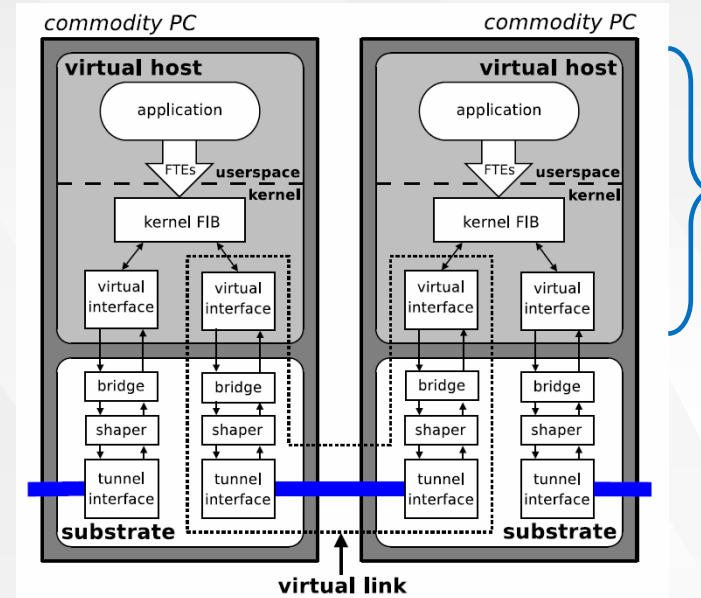
Virtual link의 효율적이며 빠른 encapsulation, multiplexing을 위해서 GRE를 활용하여 새로운 터널링 모듈인 EGRE을 구현

Terminate tunnels in the "root context"

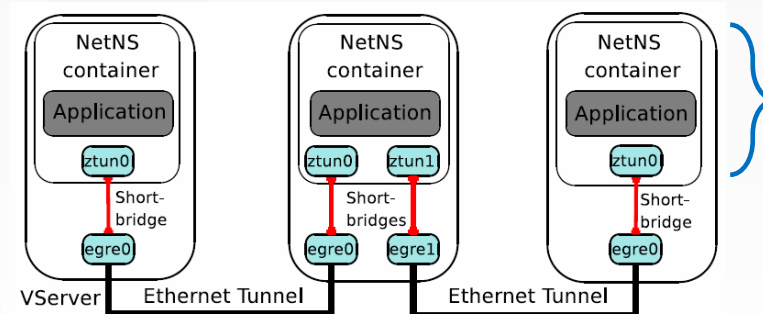
Root context에서 virtual link를 종료 시킴으로써 트래픽 제어를 보다 용이 하도록 하였다.
리눅스 트래픽 제어 모듈인 tc를 사용

Bridging

Point-to-multipoint를 위해서는 리눅스에서 지원되는 software bridging 사용
Point-to-point 연결의 성능 향상을 위해서 shortbridge를 구현



Overview of Trellis



Shortbridge



PRIVATE VINI (MYVINI)

설치 / 검증 단계 및 유의 사항

활용을 위한 운용 및 확장

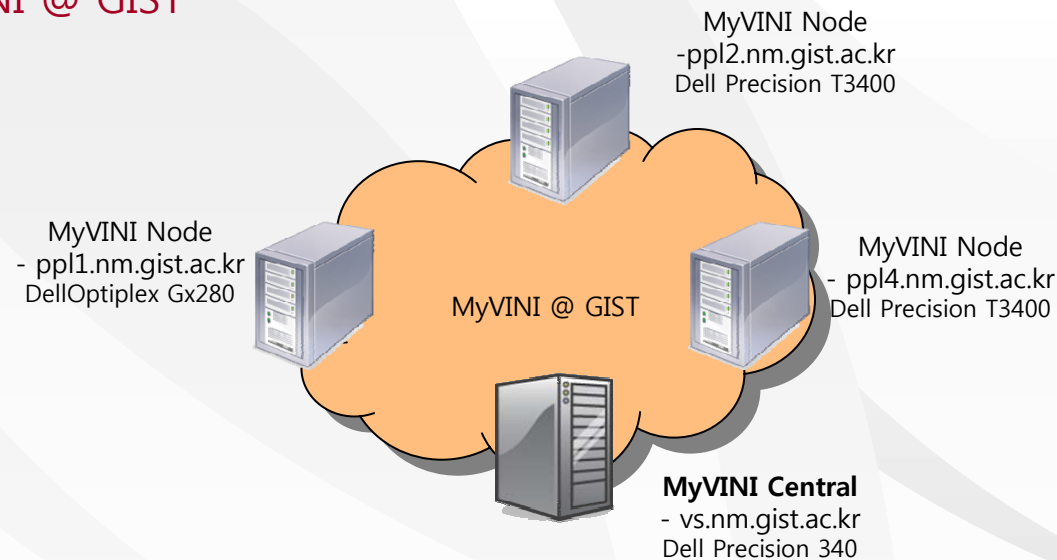


Private VINI 설치 및 검증 (1/7)

Private VINI: MyVINI

- "A standalone software distribution that allows researchers and application developers to deploy **their own virtual network hosting platforms**"
- PlanetLab (PLC)의 private 버전이 Private PlanetLab (MyPLC)인 것처럼, MyVINI는 VINI의 private 버전으로 이해할 수 있다.
- (Private PlanetLab처럼) 연구에 필요한 기능을 임의로 추가할 수 있으며, (VINI 처럼) 원하는 다양한 네트워크 환경을 꾸밀 수 있다.

예) MyVINI @ GIST



DEPT. OF INFO. & COMM., GIST



Private VINI 설치 및 검증 (2/7)

MyVINI (Trellis) 설치

■ 설치 버전

➤ VINI Trellis v0.1 kernel (2.6.22.14-vs2.3.0.32.trellis)

■ 설치한 Trellis Known issues

➤ IP 멀티캐스트 지원 안됨

➤ XORP 지원 안됨

➤ Shortbridge 대신 리눅스 bridge 지원

■ 설치 매뉴얼

➤ MyVINI installation guide

– <http://svn.planet-lab.org/wiki/MyViniInstallGuide>

➤ New MyPLC user guide

– <http://svn.planet-lab.org/wiki/MyPLCUserGuide>

➤ Vserver installation on Centos5

– <http://svn.planet-lab.org/wiki/VserverCentos>

➤ Centos5 installation

– http://wiki.linux-vserver.org/Installation_on_CentOS



Private VINI 설치 및 검증 (3/7)

MyVINI 설치 순서

- CentOS (Host OS) 설치 -> Fedora 8 (Guest OS) 설치 -> MyPLC 설치 -> Trellis 설치

MyVINI 설치 및 유의 사항

■ CentOS 설치

- ☞ 새로운 부트이미지 설치 후 재시작 할 때 새로운 부트 이미지 선택

■ Fedora 8 설치

- ☞ Host 머신에 httpd server가 동작하고 있어야 함
 - 설치 시 로컬 web server에 접속하여 필요한 데이터 수신함

■ MyPLC 설치 후 실행

- ☞ Host 머신의 httpd server의 동작을 중지시켜야 함
 - Guest 머신에서 httpd server가 동작해야 하는데 Host와 Guest 머신에서 동작하는 httpd server 간의 충돌이 있어 MyPLC 실행 시 'web server'와 'crond'가 실행되지 않음.

■ Trellis 이미지 교체

- ☞ <http://www.vini-veritas.net/boot/bootstrapfs-planetlab-i386.tar.bz2> 에서 trellis 이미지를 다운받은 후에 MyPLC가 설치된 Guest 머신의 /var/www/html/boot/bootstrapfs-planetlab-i386.tar.bz2 파일과 교체한다.

■ VINI 노드 설치

- ☞ PlanetLab 노드 설치와 동일



Private VINI 설치 및 검증 (4/7)

MyVINI 토폴로지 생성 절차

-]# make configs
 - ☞ VINI 노드들로 전송될 설정 파일이 생성된다.
-]# make sync
 - ☞ 생성된 설정 파일이 VINI 노드로 전송된다.
-]# make install
 - ☞ 필요한 패키지들이 각 VINI 노드에 설치된다.
-]# make topo
 - ☞ 설정 파일에 기초하여 가상 네트워크를 구성한다.
-]# make test
 - ☞ 설정된 가상 네트워크의 연결성을 ping으로 테스트한다.
-]# make start
 - ☞ 가상 네트워크 상에서 라우팅 소프트웨어 (예, Quagga)를 동작시킨다.
-]# make stop
 - ☞ 동작하고 있는 라우팅 소프트웨어를 중지시킨다.
-]# make teardown
 - ☞ 생성된 가상 네트워크를 없앤다

Private VINI 설치 및 검증 (5/7)

“make configs”

■ 관련된 파일

☞ Makefile

- 설정 파일 생성시 사용될 루비 파일 경로를 지정한다.
- `CONFIG= $(BASE_DIR)/config/vini.rb`

☞ Vini.rb 파일

- `###Slices`
`$iias = Slice.new('nml_2') //slice name`
`$iias.set_environment('Trellis')`
`$iias.set_key(110); //erge key`

###Nodes

- `$ppl1 = Node.new('ppl1.nm.gist.ac.kr', $iias, 'ppl1')`
- `$ppl2 = Node.new('ppl2.nm.gist.ac.kr', $iias, 'ppl2')`

###Links

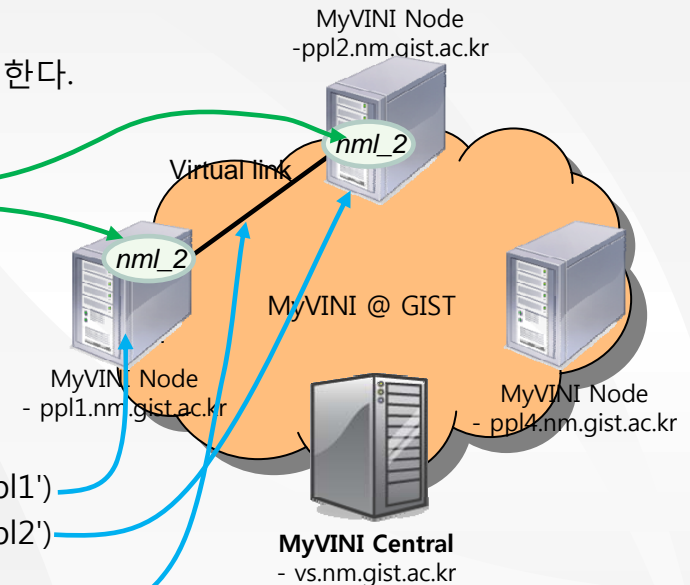
- `$l1 = Link.new($ppl1, $ppl2)`

###Additional node configurations

- `$ppl1.hostinfo("210.125.84.14")`
- `$ppl2.hostinfo("210.125.84.13")`

☞ 파일 생성

- Vini 디렉토리에 생성된 파일 저장됨



```
ace@server:~/suhyun/trellis/visp/iias/vini
[ace@server vini]$ ls
bin      nml_2@ppl1.nm.gist.ac.kr  node_list  rps     tar
hosts    nml_2@ppl3.nm.gist.ac.kr  openvpn    scripts
[ace@server vini]$
```



Private VINI 설치 및 검증 (6/7)

❏ "make topo"

■ 관련 파일: "setup-links"

■ 가능한 에러 메시지

❏ "audit_log_user_command(): Connection refused"

- 에러 메시지를 발생하나 명령어는 제대로 동작
- Fedora 8에서만 발생하는 문제로 sudo user와 연관이 있음

❏ "/vsys/setup-nat: line 14: echo: write error: File exists"

- Named pipe 관련 에러
- 에러 메시지는 발생하나 정상적으로 동작

■ 가능한 에러

❏ "SIOCADDRT: File exists"

- "make topo" 진행 시 실행되는 'setup-links'는 NAT interface를 활성화시키고 이와 관련된 routing table을 설정하다.
- 이 때 만약 기존에 동일한 이름으로 routing table에 설정이 되어 있다면 위와 같은 에러가 발생 (예, *make topo* 명령을 여러 번 실행 했을 경우)
- 문제 해결: routing table에 해당 내용이 없을 경우에 실행되도록 "setup-links"파일 수정.

❏ VINI 노드의 /etc/vini/egre-keys.txt 관련 에러

- 만약 위의 파일에 사용할 egre key 값이 없다면 가상 네트워크 인터페이스 생성이 이루어지지 않는다.



Private VINI 설치 및 검증 (7/7)

“make topo”

■ 실행 결과

```
nml_2@pp11:~  
[nml_2@pp11 ~]$ /sbin/ifconfig  
all10if0 Link encap:Ethernet HWaddr FA:41:48:A2:61:B0  
inet addr:192.168.100.2 Bcast:192.168.100.255 Mask:255.255.255.0  
UP BROADCAST RUNNING MULTICAST MTU:1458 Metric:1  
RX packets:6 errors:0 dropped:0 overruns:0 frame:0  
TX packets:6 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:384 (384.0 b) TX bytes:468 (468.0 b)  
  
eth0 Link encap:Ethernet HWaddr 00:11:43:00:1E:50  
inet addr:210.125.84.14 Bcast:210.125.84.255 Mask:255.255.255.0  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:642724 errors:0 dropped:0 overruns:0 frame:0  
TX packets:25140 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:47474587 (45.2 MiB) TX bytes:3437865 (3.2 MiB)  
Interrupt:16  
  
lo Link encap:Local Loopback  
inet addr:127.0.0.1 Mask:255.0.0.0  
UP LOOPBACK RUNNING MTU:16436 Metric:1  
RX packets:32036 errors:0 dropped:0 overruns:0 frame:0  
TX packets:32036 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:2754296 (2.6 MiB) TX bytes:2754296 (2.6 MiB)  
  
natx110 Link encap:Ethernet HWaddr 6E:18:EB:21:2B:B2  
inet addr:10.0.110.1 Bcast:10.255.255.255 Mask:255.0.0.0  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:6 errors:0 dropped:0 overruns:0 frame:0  
TX packets:19410 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:384 (384.0 b) TX bytes:815436 (796.3 KiB)  
  
nat110 Link encap:Ethernet HWaddr BE:5F:C4:EF:C8:19  
inet addr:10.0.110.2 Bcast:10.0.110.255 Mask:255.255.255.0  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:19410 errors:0 dropped:0 overruns:0 frame:0  
TX packets:6 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:543696 (530.9 KiB) TX bytes:468 (468.0 b)  
  
[nml_2@pp11 ~]$
```

```
nml_2@pp11:~  
[nml_2@pp11 ~]$ /sbin/route  
Kernel IP routing table  
Destination Gateway Genmask Flags Metric Ref Use Iface  
192.168.100.0 * 255.255.255.0 U 0 0 0 a110if0  
192.168.100.0 * 255.255.255.0 U 0 0 0 a110if0  
210.125.84.0 * 255.255.255.0 U 0 0 0 eth0  
10.0.110.0 * 255.255.255.0 U 0 0 0 nat110  
10.0.110.0 * 255.255.255.0 U 0 0 0 nat110  
link-local * 255.255.0.0 U 0 0 0 eth0  
10.0.0.0 * 255.0.0.0 U 0 0 0 natx110  
default 10.0.110.1 0.0.0.0 UG 0 0 0 natx110  
default 210.125.84.1 0.0.0.0 UG 0 0 0 eth0  
[nml_2@pp11 ~]$ /sbin/ifconfig
```

Private VINI 운용 및 확장 (1/3)

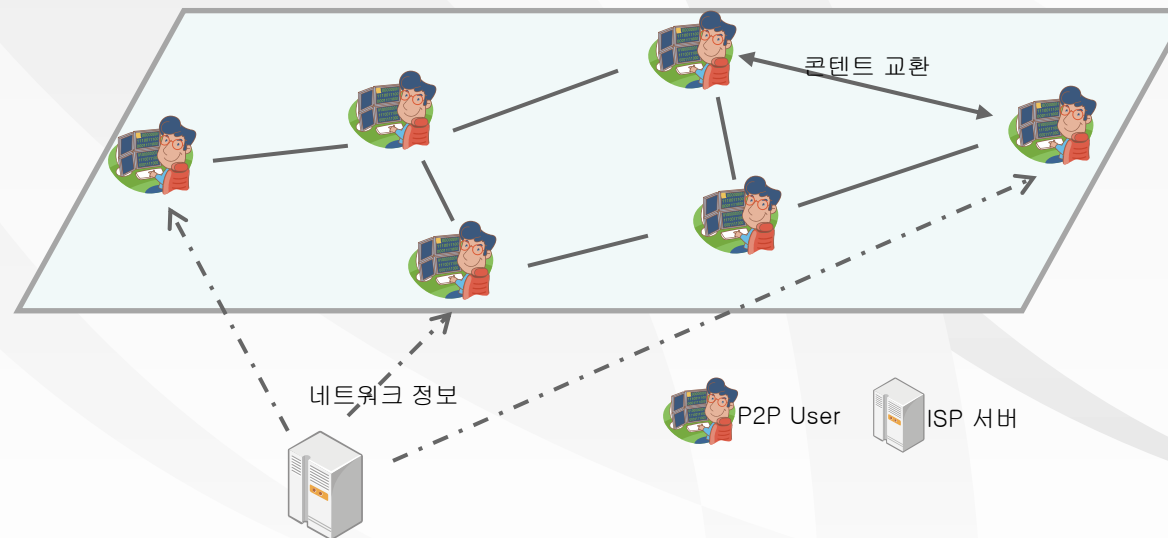
활용 방안 예 - Dynamic IPS (Informed Peer Selection)

Informed Peer Selection

☞ P2P 사용자가 ISP가 제공하는 네트워크 정보에 기초하여 통신할 상대를 선택함으로써, ISP의 운용 비용을 절감시키고 P2P 사용자의 성능을 향상시키는 방법

Dynamic IPS

☞ IPS에 기초하여 선택된 상대와 통신하는 동안 네트워크 상황 변화에 대응하여 P2P 사용자의 품질을 보장하기 위해 P2P 사용자 간의 통신 경로를 변경하는 방법

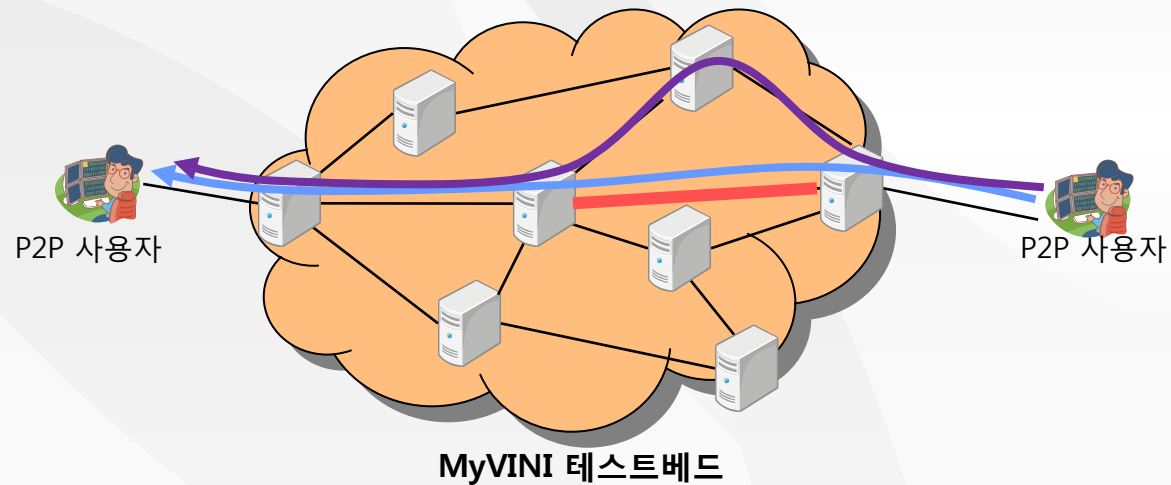


Private VINI 운용 및 확장 (2/3)

동적인 플로우 제어 기능 추가

■ 목표

- 미리 설정한 라우팅 프로토콜에 의한 라우팅 뿐 아니라, 연구자 임의로 자신이 원하는 특정 플로우를 제어할 수 있는 기능을 추가



■ 요구 사항

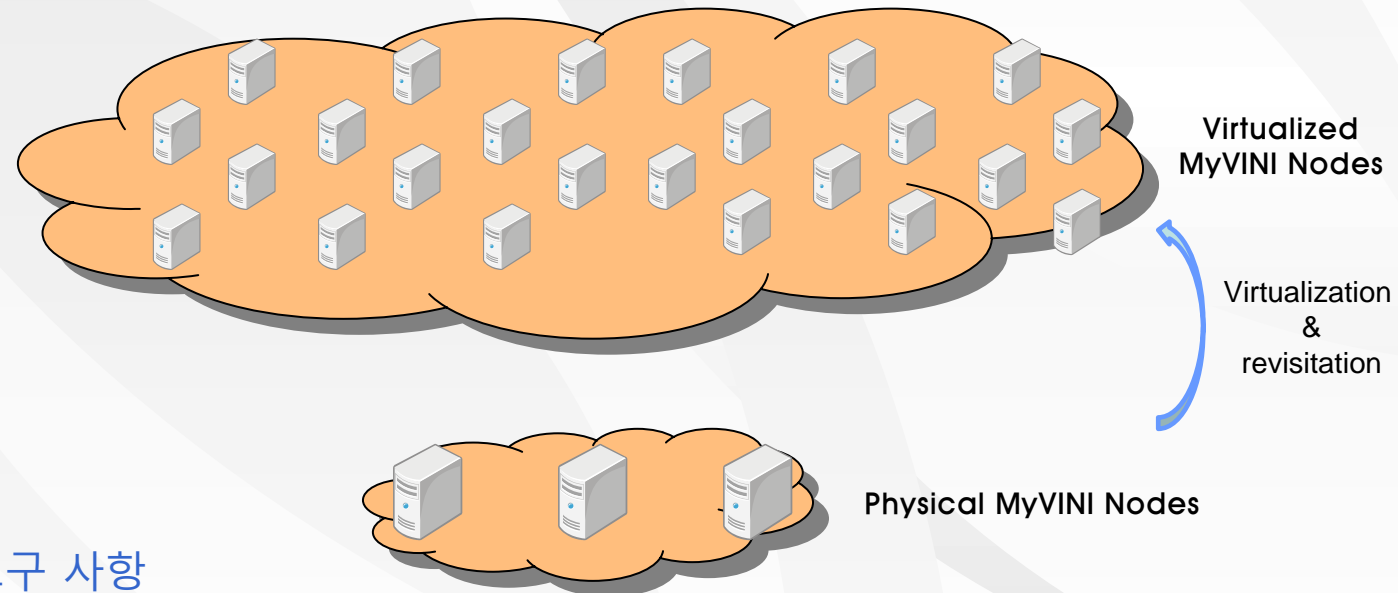
- VINI를 활용한 네트워크 이벤트 발생 (VINI 지원)
- VINI에서 가상 링크의 네트워크 상황을 관찰할 수 있는 기능 추가
 - 예) S³(Scalable Sensing Service): E2E latency, E2E available bandwidth, loss rate ...
- 포워딩 테이블을 동적으로 변경할 수 있는 기능 추가
 - 예) "route add/delete ~~" 명령을 통한 라우팅 테이블 변경

Private VINI 운용 및 확장 (3/3)

대규모의 테스트베드 구축

목표

- “revisitation” 개념 적용을 통해 적은 수의 노드로 대규모의 테스트베드를 구축
- revisitation: allows a physical in the underlying infrastructure to host multiple virtual nodes of a single virtual network



요구 사항

- 가상 호스트를 연결하는 가상 링크의 네트워크 설정 (예, 지연시간, 패킷 손실율..)
- 동일 노드에서 지원되는 가상 호스트 간의 성능 분리 (isolation) 확인



Q & A

Thanks!!



DEPT. OF INFO. & COMM., GIST

