

오픈 SmartX 플랫폼을 위한 SmartX Box 기반 KOREN L3 연동

Document No. SmartX #2

Version 1.0

Date 2015-12-11

Author(s) GIST Team

■ 문서의 연혁

버전	날짜	작성자	비고
초안 - 0.1	2015. 10. 16	GIST OF@KOREN 팀	
1차 수정	2015. 11. 11	GIST OF@KOREN 팀	
2차 수정	2015. 11. 25	GIST OF@KOREN 팀	
최종	2015. 12. 11	GIST OF@KOREN 팀	
1.0	2015. 12. 11		

본 연구는 한국정보화진흥원(NIA)의 미래네트워크연구시험망(KOREN) 사업 지원과제의 연구결과로 수행되었음 (15-951-00-001)

This research was one of KOREN projects supported by National Information Society Agency (15-951-00-001)

Contents

SmartX #2. 오픈 SmartX 플랫폼을 위한 SmartX Box 기반 KOREN L3 연동

1. SmartX Box 및 SmartX Playground 개요	5
1.1. 목적 및 개요	5
1.2. SmartX Playground	6
1.3. SmartX 기반 KOREN L3 연동의 필요성	6
2. SmartX Box 기반 KOREN L3 연동	7
2.1. SmartX Box 기반 KOREN L3 연동 개요	7
2.2. SmartX Box 기반 KOREN L3 연동 구현	7
2.3. SmartX Box 기반 KOREN L3 연동 동작과정 설명	8
2.4. SmartX Box 기반 KOREN L3 연동의 개선계획	9
3. SmartX Box 기반 KOREN L3 연동의 설정방법	11
3.1. VyOS 개요 및 설정	11
3.1.1. VyOS 개요	11
3.1.2. VyOS 설정	11
3.1.3. Site-to-site VPN tunnel 설정	15
3.2. OpenStack instance to vRouter 개요 및 설정	18
3.2.1. OpenStack instance to vRouter 연결 개요	18
3.2.2. OpenStack instance to vRouter 연결 상세	18
4. End to End Security 검증	23
4.1. End to End Security 연결 검증	23

그림 목차

그림 1 융합형 SmartX Box 및 Box/Inter-Connect/Function의 3대 요소	5
그림 2 OF@KOREN Playground 구성도	6
그림 3 SmartX Box/Switch를 활용한 KOREN L3 연동 개념도	7
그림 4 KOREN L3 연동을 검증할 위한 구현 구조	8
그림 5 L3 VPN이 적용된 OpenStack Cloud의 모습	9
그림 6 End-to-End Security	10
그림 7 vRouter 설치 과정 1	12
그림 8 vRouter 설치 과정 2	12
그림 9 vRouter 설치 과정 3	12
그림 10 vRouter 설치 과정 4	13
그림 11 vRouter 설치 과정 5	13
그림 12 vRouter의 네트워크 인터페이스 추가	14
그림 13 GIST SANDBOX br-ex의 네트워크 설정	14
그림 14 GIST SANDBOX 내부 vRouter의 네트워크 인터페이스 설정	14
그림 15 vpn 인터페이스 설정	15
그림 16 GIST SANDBOX 내부 vRouter의 IKE group 설정	16
그림 17 GIST SANDBOX 내부 vRouter의 ESP group 설정	16
그림 18 사이트 간 L3 IPsec VPN tunnel 연동	17
그림 19 GIST SANDBOX 내부 vRouter의 IPsec VPN tunnel 설정	17
그림 20 두 사이트의 vRouter간 IPsec VPN tunnel	18
그림 21 가상 스위치 연결	20
그림 22 OpenStack flat network 생성	20
그림 23 flat network에 서브넷 생성	21
그림 24 vRouter에 새로운 네트워크 인터페이스 추가	21
그림 25 vRouter의 네트워크 인터페이스 설정 화면	22
그림 26 SANDBOX 내부 OpenStack instance와 vRouter의 연결	22
그림 27 OpenStack 네트워크 토폴로지	23
그림 28 POSTECH SmartX Type C 내부 vRouter 콘솔	24

표 목차

표 1 GIST SANDBOX 1 및 POSTECH Type C Box 사양	11
--	----

SmartX #2. 오픈 SmartX 플랫폼을 위한 SmartX Box 기반 KOREN L3 연동

1. SmartX Box 및 SmartX Playground 개요

1.1. 목적 및 개요

- 본 문서에서는 현재 국내 여러 사이트의 SmartX Type C Box를 연동하여 운용 중인 OF@KOREN Playground를 위한 KOREN OpenStack Cloud 인프라의 L3 연동 확장 검증을 위한 구조 설계 및 구현 그리고 추후 개선 계획에 대해 기술한다. 기존 Underlay Network 위에 VXLAN 방식의 터널링 기법을 통해 확보되는 분산 OpenStack Cloud의 종단 간 네트워킹 연결성을 SmartX Box 내부에 가상 라우터를 배치함으로써 L3 네트워크 방식의 연결성을 확보할 수 있다.
- 본 기술문서의 L3 연동의 대상이 되는 그림 1과 같은 컴퓨팅/네트워킹/스토리지 자원의 초융합형 자원 Box를 SmartX Box라 지칭한다. 이러한 초융합형 자원 Box들을 통해 OpenStack Cloud를 구성함과 동시에 Box 내부에 가상라우터/스위치를 배치해 모든 Box와 연결된 중앙 집중형 Box인 컨트롤 타워에서 각 사이트의 설정 및 관리를 할 수 있다. 그림 1의 중심부에 위치하는 Box는 내부에 컴퓨팅/스토리지/네트워킹의 요소를 하나로 묶는 초융합형 자원 박스를 나타낸다.

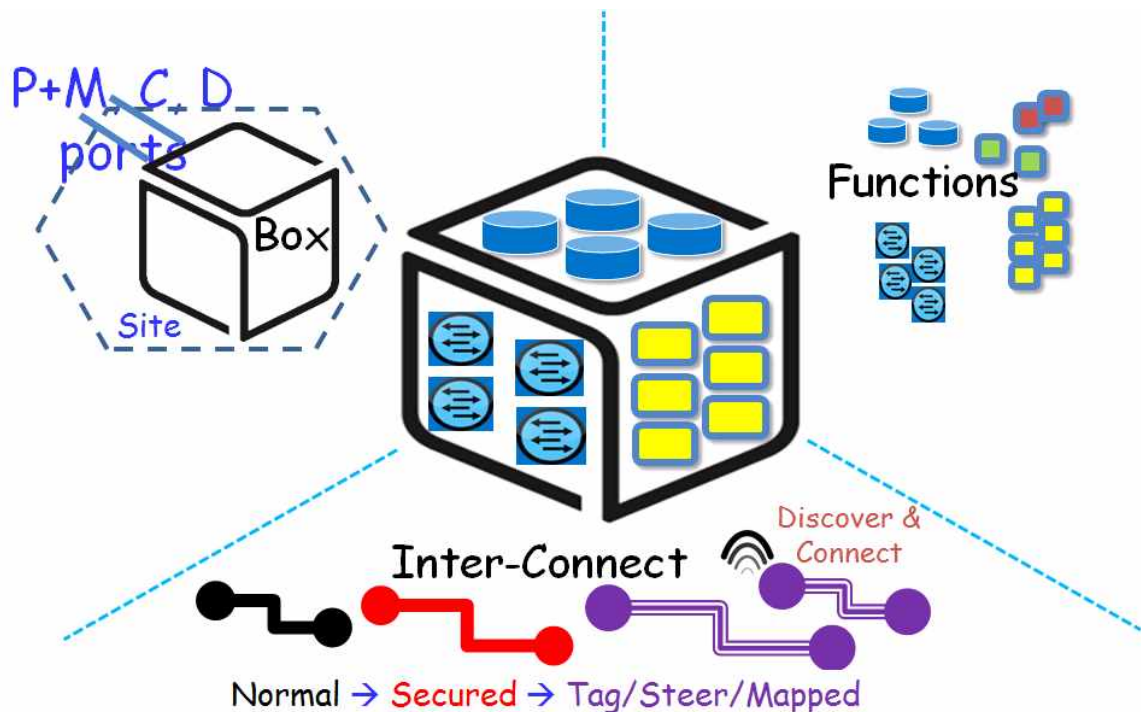


그림 1 융합형 SmartX Box 및 Box/Inter-Connect/Function의 3대 요소

1.2. SmartX Playground

- 본 기술문서의 KOREN L3 연동 대상인 SmartX Box들을 연동해 구현한 OpenStack Cloud 인프라를 사용자가 원하는 실험을 할 수 있도록 제공하고, 이를 사용자들을 위한 SmartX Playground라 한다.
- 이러한 SmartX Playground는 사용자들이 실험을 위해 필요로 하는 컴퓨팅/스토리지/네트워킹 자원을 요구에 맞게 동적이고 신속, 유연하게 제공할 수 있어야 한다.

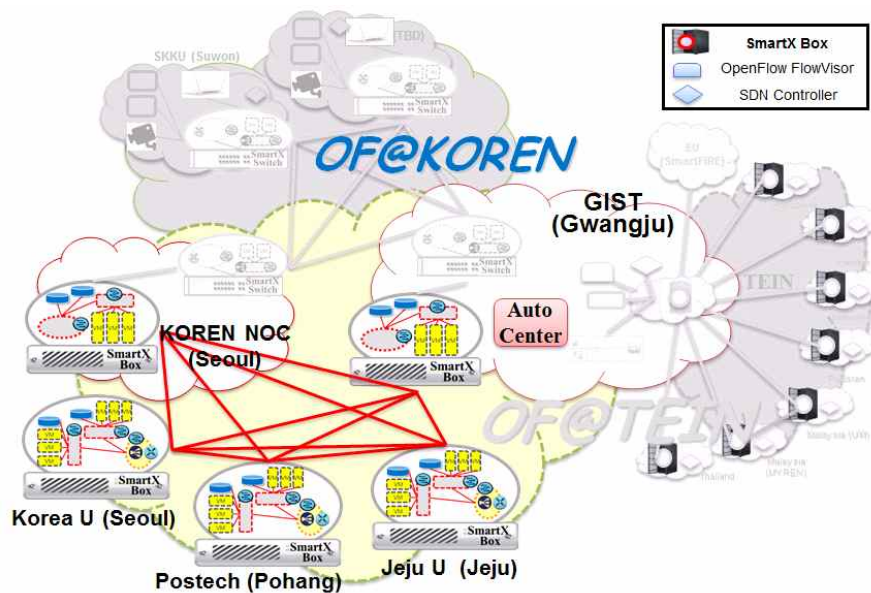


그림 2 OF@KOREN Playground 구성도

- 현재 이러한 환경을 위해 광주과학기술원의 NetCS 연구실을 중심으로 국내 5개 사이트를 연동하여 OF@KOREN SmartX Playground를 제공하고 있다.

1.3. SmartX 기반 KOREN L3 연동의 필요성

- 기존 분산 OpenStack Cloud 인프라의 종단 간 네트워킹 연결성은 Underlay Network 위에 VXLAN 방식의 터널링 기법을 통해 가상 오버레이 네트워킹의 형태로 확보되고 있다. 이러한 오버레이 네트워킹 방식의 연동은 Underlay Network 환경을 세부적으로 고려하지 않고 인프라 구성을 쉽고 빠르게 할 수 있다는 장점을 가지고 있으나, 적용된 터널링 기법의 한계로 인해 네트워킹 성능, 확장성 제한, 추가 장비 필요 등의 단점들을 가지고 있다. 이러한 문제점을 해결하기 위하여 SmartX Box 내부에 가상 라우터를 배치해, 라우팅 기반의 L3 네트워킹 방식으로 SmartX Box를 활용한 다지점 OpenStack Cloud의 종단 간 연결성을 확보한다.

2. SmartX Box 기반 KOREN L3 연동

2.1. SmartX Box 기반 KOREN L3 연동 개요

- o SmartX Box 기반 KOREN L3 연동은 각 사이트에 위치한 SmartX Box 내부에 가상라우터(vRouter)를 배치해 L3 기반의 사이트 간 연결성을 제공하는 것을 목적으로 한다. 또한 IPsec VPN tunnel을 활용해 사이트 간 연동에 security 측면에 기여할 수 있다.

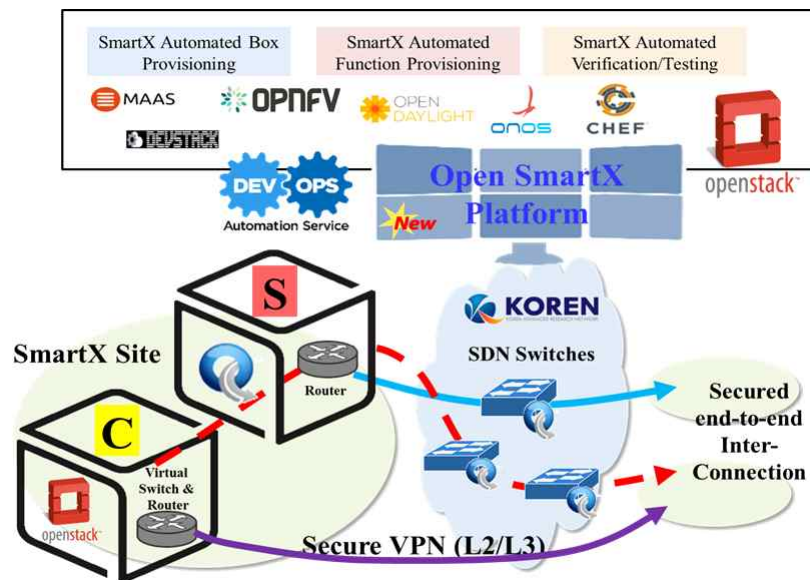


그림 3 SmartX Box/Switch를 활용한 KOREN L3 연동 개념도

- o 본 기술문서는 기존 VXLAN 터널링 기반의 L2 수준 연동이 적용된 OpenStack Cloud를 L3 수준의 연동을 검증함으로써 기존 L2 수준 연동의 한계점을 극복하고 사이트 간 안전한 연동 목표로 한다.

2.2. SmartX Box 기반 KOREN L3 연동 구현

- o 본 기술문서에서는 SmartX Box 기반 KOREN L3 연동을 검증하기 위해 아래 그림 3과 같은 구조를 구현하였다.

Test Environment

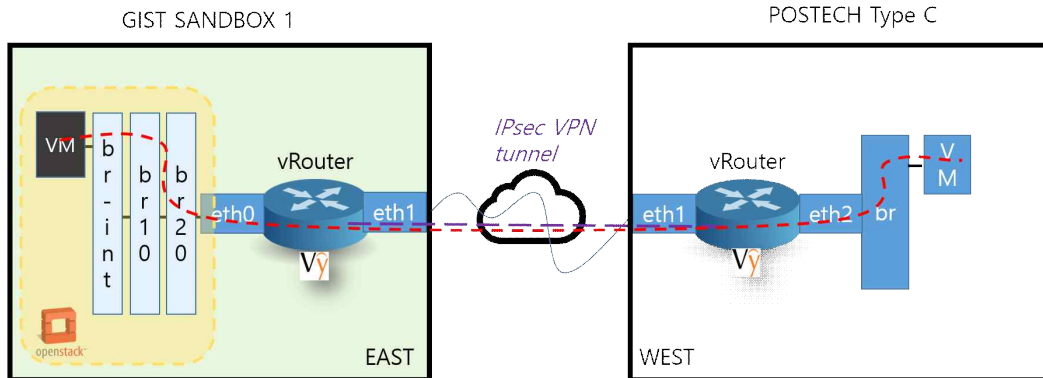


그림 4 KOREN L3 연동을 검증할 위한 구현 구조

- o SmartX Type C Box들로 구성된 KOREN OpenStack 클라우드 인프라의 사이트 간 L3 연동 실험을 위해 GIST의 SANDBOX1과 POSTECH의 SmartX Type C Box가 사용되었다. 사이트 간 L3 연동을 위해 Box 내부에 가상 라우터/스위치가 배치된다.
- o SmartX Box 내부에 가상 라우터를 배치할 경우 두 개의 네트워크 인터페이스를 필요로 하는데, 하나의 네트워크 인터페이스는 Box 내부의 OpenStack 네트워크와 연결되어 OpenStack의 instance들의 네트워크와 같은 대역의 IP 주소를 갖는다. 다른 네트워크 인터페이스는 물리적 네트워크 인터페이스와 연결되어, 퍼블릭 아이피를 가지며 다른 사이트의 가상 라우터와 연결되어 L3 VPN tunnel을 형성한다. 본 기술문서의 검증에서 가상 라우터를 만들기 위하여 오픈 소스 VyOS를 사용한다. 구현에는 오픈소스 VyOS가 자체적으로 지원하는 L3 기반의 IPsec site-to-site vpn tunnel 기술이 사용된다. 또한 가상 라우터에 라우팅 테이블 설정을 통해 패킷들이 이동할 경로를 지정할 수 있다.
- o SmartX Box 내부의 OpenStack과 가상 라우터를 연결하는 데 있어서 필요한 가상 스위치를 구현하는 데 있어서 오픈 소스 Open vSwitch를 활용한다. OpenStack에서 나오는 패킷들이 지나가는 br-int 브릿지에 Open vSwitch를 활용해 구축한 가상 스위치를 연결하여 모든 패킷들이 가상 라우터의 네트워크 인터페이스로 지나가도록 구현하여 가상 라우터가 OpenStack instance들의 라우팅을 담당해 다른 사이트에 존재하는 SmartX Box 들과 연동할 수 있다.

2.3. SmartX Box 기반 KOREN L3 연동 동작과정 설명

- o SmartX Box 기반 KOREN L3 연동의 동작과정은 크게 세 단계로 나눌 수 있다.

- o 첫 번째 단계는 OpenStack 내부의 인스턴스에서 발생한 패킷이 동일 Box 내부의 가상 라우터로 흘러가는 과정이다. OpenStack의 네트워크와 가상 라우터가 같은 대역의 사설 IP를 가지고 있고, 연결된 가상 스위치를 통해 패킷이 이동한다.
- o 가상 라우터에 패킷이 이동하면 두 번째 단계가 시작된다. 설정된 라우팅 테이블과 기존에 구현 되어 있는 IPsec vpn tunnel을 통해 패킷들이 안전하게 보호된 상태로 다른 사이트에 위치한 가상 라우터로 이동한다.
- o 세 번째 단계는 다른 사이트에서 가상 라우터에 도착한 패킷들이 OpenStack instance로 이동하는 단계이다. 가상 라우터에 설정 된 라우팅 테이블에 따라 패킷이 이동하게 된다.

2.4. SmartX Box 기반 KOREN L3 연동의 개선계획

- o 본 기술문서에서는 SmartX Box 기반 L3 연동 검증하기 위한 구현을 실시하였다. 추후 L3 수준 연동을 실제 KOREN OpenStack Cloud 인프라에 적용시킨 후 SmartX Server-Switch를 통합하는 L3 네트워크로 확장할 수 있을 것이다. 또한 SDN 제어기를 통해 통합된 L3 네트워크를 동적으로 유연하게 제어 가능하도록 개선될 수 있다. 또한 L3 수준 연동에 사용되는 가상 스위치와 가상 라우터를 SDN 제어기에 연결하여 KOREN의 네트워크에 소프트웨어 정의 네트워킹을 적용할 수 있을 것이다.

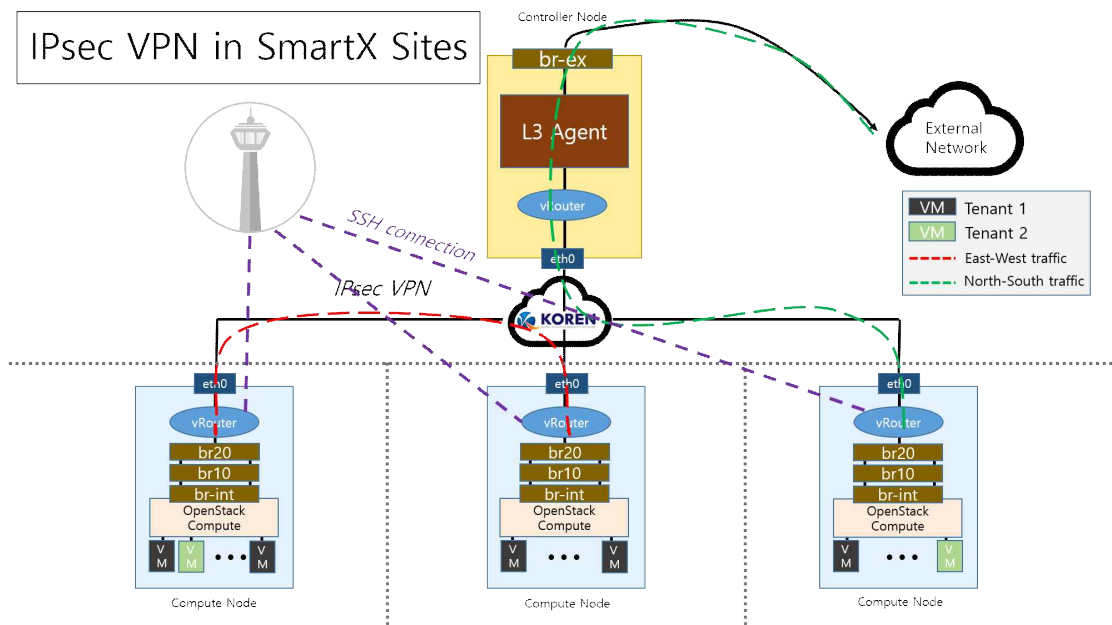


그림 5 L3 VPN이 적용된 OpenStack Cloud의 모습

- o 또한 더 나아가 End-to-End security의 측면에서 OpenStack Cloud만이 아닌 IoT 디바이스에 직접 연결되어 있는 μ -cloud에도 가상 라우터와 가상 스위치를 활용한 L3 수준 연동이 가능할 것이다.

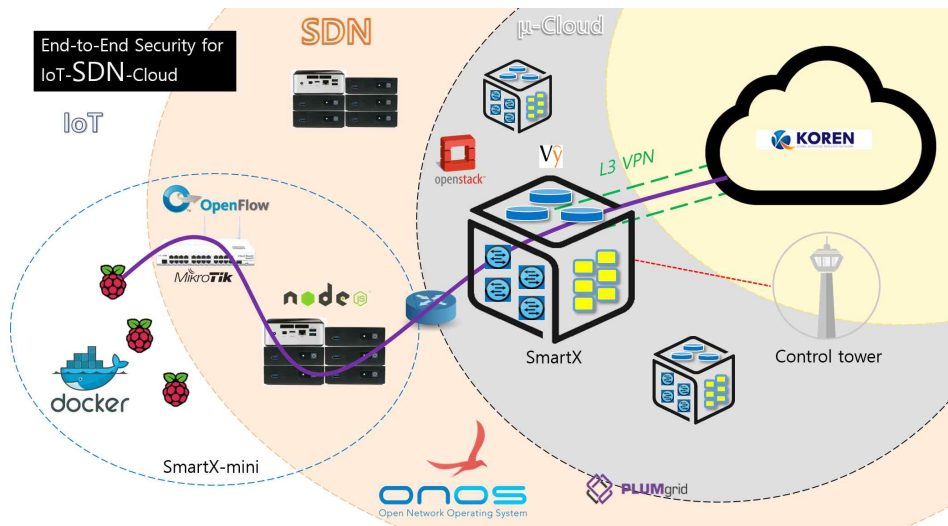


그림 6 End-to-End Security

3. SmartX Box 기반 KOREN L3 연동의 설정방법

3.1. VyOS 개요 및 설정

3.1.1. VyOS 개요

- o VyOS는 리눅스 기반의 네트워크 운영체제로, 소프트웨어 기반 네트워크 라우팅과 방화벽, VPN 등을 제공해준다 [1]. VyOS는 Vyatta의 포크 버전으로, Vyatta의 경우 2006년부터 무료로 사용이 가능하였으나, 2012년 Brocade Communications Systems에 의해 인수되었다. 이러한 이유로 본 구현에서는 오픈 소스인 VyOS를 이용해 vRouter를 구현하였다. 현재 VyOS는 2015년 8월 기준 1.1.6 버전이 배포되는 중이다.
- o VyOS는 Xen, VMware, KVM 등의 가상머신 환경에서 돌아가게 된다. 최소사양은 512MB RAM과 2GB 저장 공간이다. VyOS는 자체적으로 IPsec VPN 기능을 지원해 이를 통해 사이트 간 안전한 L3 VPN tunnel을 구현할 수 있다. VyOS는 공식 홈페이지에서 이미지를 배포하고 있다.

3.1.2. VyOS 기본 설정

- o VyOS를 통한 vRouter 구현은 KVM을 통해 이루어진다. 구현 환경은 다음과 같다.

표 1. GIST SANDBOX 1 및 POSTECH Type C Box 사양

	GIST SANDBOX 1	POSTECH Type C
CPU	Intel® Xeon E5-2630, 6 cores	Intel® Xeon E5-2690, 10*2 cores
RAM	16GB	96GB
Storage	120GB HDD	1.3GB SSD
OS	Ubuntu 14.04.3 LTS	Ubuntu 14.04.2 LTS

VyOS vRouter의 설치 과정은 다음과 같다. 이번 구현에서 연결할 GIST의 Sandbox와 POSTECH의 SmartX Type C Box 상에서 KVM을 통해 가상머신 형태로 VyOS vRouter를 생성한다.

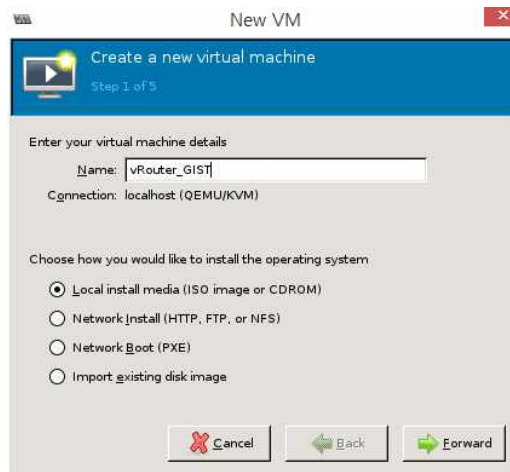


그림 7 vRouter 설치 과정 1

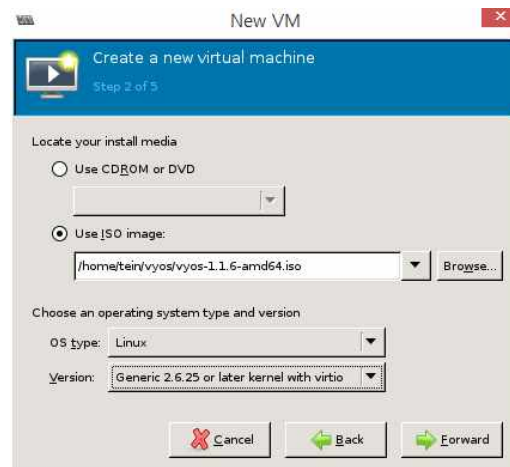


그림 8 vRouter 설치 과정 2

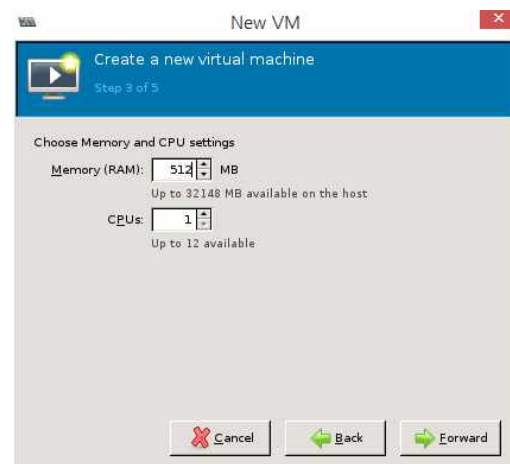


그림 9 vRouter 설치 과정 3

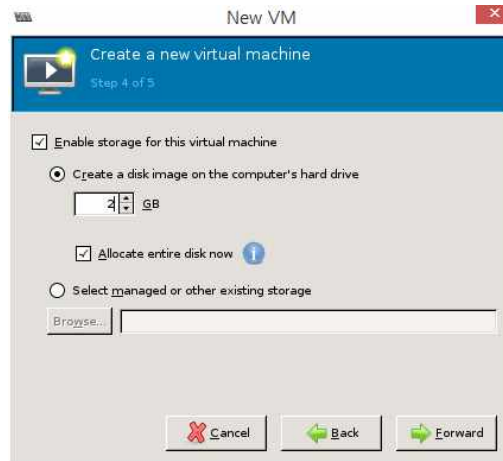


그림 10 vRouter 설치 과정 4



그림 11 vRouter 설치 과정 5

vRouter의 초기 아이디/비밀번호는 vyos/vyos이다. vRouter가 설치된 가상머신을 구동 후 다음 커맨드를 통해 이미지를 설치한다.

```
$ install image
```

두 머신에 vRouter를 설치한 후, 두 사이트의 머신 간의 L3 VPN 연동에 앞서 vRouter의 퍼블릭 아이피 설정을 위한 network interface를 추가한다. Sandbox 환경의 경우 br-ex가 퍼블릭 아이피를 가지고 있으므로 다음과 같이 가상 네트워크 인터페이스를 추가한다.

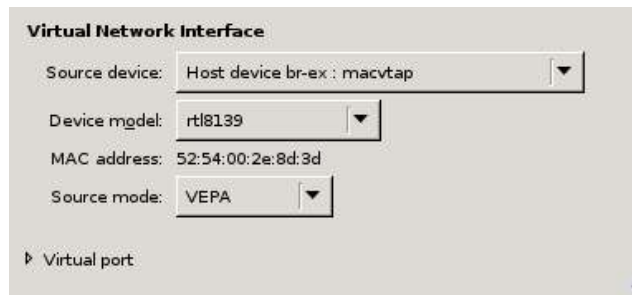


그림 12 vRouter의 네트워크 인터페이스 추가

network interface를 추가한 후 \$ reboot 명령어를 통해 가상머신을 재부팅한다. br-ex의 ip 설정은 그림 13과 같다. vRouter의 IP를 br-ex와 같은 대역으로 설정한다.

```
br-ex  Link encap:Ethernet  HWaddr e4:1f:13:ec:a2:92
       inet addr:210.114.90.177 Bcast:210.114.90.255 Mask:255.255.255.0
       inet6 addr: fe80::6c61:aff:fe49:f967/64 Scope:Link
       UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
       RX packets:6366776 errors:0 dropped:1375 overruns:0 frame:0
       TX packets:2931543 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:0
       RX bytes:1031106067 (1.0 GB)  TX bytes:7259268321 (7.2 GB)
```

그림 13 GIST SANDBOX br-ex의 네트워크 설정

```
$ configure
# set interfaces ethernet eth1 address 210.114.90.80
# set system gateway-address 210.114.90.254
# set system name-server 8.8.8.8
# commit
# save
```

```
vyos@vyos# set interfaces ethernet eth0 address 210.114.90.80/24
[edit]
vyos@vyos# show interfaces
  ethernet eth0 {
+   address 210.114.90.80/24
    hw-id 52:54:00:eb:39:a0
  }
  loopback lo {
  }
[edit]
vyos@vyos#
```

그림 14 GIST SANDBOX 내부 vRouter의 네트워크 인터페이스 설정

3.1.3. Site-to-site VPN tunnel 설정

- o vRouter의 기본적인 설정이 완료된 후에는 두 사이트의 vRouter를 연결하기 위한 vpn tunnel의 설정이 필요하다. 우선 GIST의 SANDBOX 내부의 vRouter를 설정한다.

- vpn 인터페이스 설정

두 사이트 간의 VPN tunnel을 구축하기 위한 vpn 인터페이스를 설정한다. 다음 명령어를 통해 vpn 인터페이스의 설정이 가능하다.

\$ configure

set vpn ipsec ipsec-interfaces interface eth1

commit

```
vyos@vyos:~$ configure
[edit]
vyos@vyos# set vpn ipsec ipsec-interfaces interface eth0
[edit]
vyos@vyos# show vpn ipsec ipsec-interfaces
+interface eth0
[edit]
vyos@vyos# _
```

그림 15 vpn 인터페이스 설정

- IKE group 설정

vpn 인터페이스 설정을 한 후에는 IPsec VPN tunnel을 위한 IKE group의 설정이 필요하다. 설정 방법은 다음과 같다.

\$ configure

set vpn ipsec ike-group IKE-1W proposal 1

set vpn ipsec ike-group IKE-1W proposal 1 encryption aes256

set vpn ipsec ike-group IKE-1W proposal 1 hash sha1

set vpn ipsec ike-group IKE-1W proposal 2 encryption aes128

set vpn ipsec ike-group IKE-1W proposal 2 hash sha1

set vpn ipsec ike-group IKE-1W lifetime 3600

```
vyos@vyos# show vpn ipsec ike-group IKE-1W
+lifetime 3600
+proposal 1 {
+  encryption aes256
+  hash sha1
+}
+proposal 2 {
+  encryption aes128
+  hash sha1
+}
[edit]
vyos@vyos# _
```

그림 16 GIST SANDBOX 내부 vRouter의 IKE group 설정

- ESP group 설정

IKE group 설정 후에는 ESP group의 설정이 필요하다. 설정 방법은 다음과 같다.

\$ configure

```
# set vpn ipsec esp-group ESP-1W proposal 1
# set vpn ipsec esp-group ESP-1W proposal 1 encryption aes256
# set vpn ipsec esp-group ESP-1W proposal 1 hash sha1
# set vpn ipsec esp-group ESP-1W proposal 2 encryption 3des
# set vpn ipsec esp-group ESP-1W proposal 2 hash md5
# set vpn ipsec esp-group ESP-1W lifetime 1800
```

```
vyos@vyos# show vpn ipsec esp-group ESP-1W
+lifetime 1800
+proposal 1 {
+  encryption aes256
+  hash sha1
+}
+proposal 2 {
+  encryption 3des
+  hash md5
+}
[edit]
vyos@vyos# _
```

그림 17 GIST SANDBOX 내부 vRouter의 ESP group 설정

- Site-to-site vpn tunnel 설정

위 설정들이 완료된 후 site-to-site vpn tunnel의 설정을 한다. 이 때 구현 환경은 다음 그림 23 과 같다.

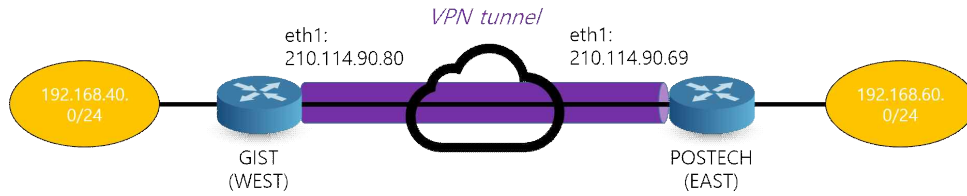


그림 18 사이트 간 L3 IPsec VPN tunnel 연동

site-to-site vpn tunnel의 설정 방법은 다음과 같다.

```
$ configure
# set vpn ipsec site-to-site peer 210.114.90.69 authentication mode
pre-shared-secret
# edit vpn ipsec site-to-site peer 210.114.90.69
# set authentication pre-shared-secret test_key_1
# set default-esp-group ESP-1W
# set ike-group IKE-1W
# set local-address 210.114.90.80
# set tunnel 1 local prefix 192.168.40.0/24
# set tunnel 1 remote prefix 192.168.60.0/24
# top
# commit
```

```
vyos@vyos# show vpn ipsec site-to-site peer 210.114.90.69
authentication {
    mode pre-shared-secret
    pre-shared-secret test_key_1
}
default-esp-group ESP-1W
ike-group IKE-1W
local-address 210.114.90.80
tunnel 1 {
    local {
        prefix 192.168.40.0/24
    }
    remote {
        prefix 192.168.60.0/24
    }
}
[edit]
vyos@vyos#
```

그림 19 GIST SANDBOX 내부 vRouter의 IPsec VPN tunnel 설정

- o 위의 설정들과 마찬가지로 POSTECH에 위치한 SmartX Type C Box에도 같은 방법의 설정을 해 준다. 이후 IPsec VPN tunnel이 정상적으로 작동하는 것을 확인할 수 있다.

```
vyos@vyos:~$ show vpn ipsec sa
```

Peer ID / IP		Local ID / IP							
-----		-----							
210.114.90.69		210.114.90.80							
-----		-----							
Tunnel	State	Bytes Out/In	Encrypt	Hash	NAT-T	A-Time	L-Time	Proto	
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
1	up	0.0/0.0	aes256	sha1	no	841	1800	all	

그림 20 두 사이트의 vRouter간 IPsec VPN tunnel

3.2. OpenStack instance to vRouter 개요 및 설정

3.2.1. OpenStack instance to vRouter 개요

- o 각 사이트의 vRouter간의 vpn tunnel이 구현된 이후에는 OpenStack 내부의 instance와 vRouter의 연결이 필요하다. 본 구현에서는 OpenStack instance들과 같은 머신 내부의 vRouter를 연결하기 위해 Open vSwitch를 사용한다. Open vSwitch는 소프트웨어 구현 분산 가상 멀티 레이어 스위치를 위한 오픈 소스이다. [2] 본 도구를 활용하여 가상 스위치를 통해 OpenStack의 instance의 패킷들이 vRouter로 갈 수 있도록 설정한다.

3.2.2. OpenStack instance to vRouter 연결 상세

- o OpenStack의 인스턴스들을 vRouter와 연결하기 위해 가상 스위치가 요구된다. 이러한 가상 스위치를 만들기 위해 Open vSwitch가 사용된다. 이번 구현에서는 OpenStack의 네트워크와 vRouter 사이에 두 개의 가상 스위치를 만든다. SANDBOX 내부에서 Open vSwitch를 통한 가상 스위치 생성과 연결 방법은 다음과 같다.

```
# ovs-vsctl add-br br10
# ovs-vsctl add-br br20
# ifconfig br10 up
# ifconfig br20 up
# ovs-vsctl add-port br10 br2010
# ovs-vsctl add-port br20 br1020
# ovs-vsctl set interfaces br2010 type=patch
# ovs-vsctl set interfaces br1020 type=patch
# ovs-vsctl set interfaces br2010 options:peer=br1020
```

```
# ovs-vsctl set interfaces br1020 options:peer=br2010
```

- o OpenStack의 네트워크를 flat 네트워크로 설정한다. 방법은 다음과 같다. 우선 OpenStack Neutron의 설정 파일을 아래와 같이 수정한다.

```
# nano etc/neutron/plugins/ml2/ml2_conf.ini
```

```
[ml2]
```

```
type_drivers = flat
```

```
tenant_networks_types = flat
```

```
mechanism_drivers = openvswitch, l2population
```

```
[ml_type_flat]
```

```
flat_networks = p10
```

```
[ovs]
```

```
local_ip = 172.30.90.10
```

```
bridge_mappings = p10:br10
```

```
[agent]
```

```
l2_population = True
```

```
enable_distributed_routing = True
```

- o 설정파일의 수정이 끝나면 새로운 설정의 적용을 위해 모든 neutron 서비스의 재시작을 해준다.

```
# service nova-api restart
```

```
# service neutron-server restart
```

```
# service openvswitch-switch restart
```

```
# service neutron-plugin-openvswitch-agent restart
```

```
# service neutron-l3-agent restart
```

```
# service neutron-dhcp-agent restart
```

```
# service neutron-metadata-agent restart
```

이후 아래 명령어를 통해 OpenStack에 flat network를 생성한다.

```
# neutron net-create \
```

```
--tenant-id 2e05722637b64192aede070137aeab22 \
--shared \
--provider:network_type flat \
--provider:physical_network p10 \
p10
```

위 명령어를 통해 네트워크를 생성하면 아래 그림과 같이 가상 스위치의 연결과, OpenStack에 flat network가 생성되었음을 확인할 수 있다.

```
Bridge "br20"
  Port "br1020"
    Interface "br1020"
      type: patch
      options: {peer="br2010"}
  Port "br20"
    Interface "br20"
      type: internal
Bridge "br10"
  Port "br2010"
    Interface "br2010"
      type: patch
      options: {peer="br1020"}
  Port "br10"
    Interface "br10"
      type: internal
  Port "phy-br10"
    Interface "phy-br10"
      type: patch
      options: {peer="int-br10"}
ovs_version: "2.3.2"
```

그림 21 가상 스위치 연결

Field	Value
admin_state_up	True
id	d2a93274-316e-40c0-b11b-b8cbb71654a2
mtu	0
name	p10
provider:network_type	flat
provider:physical_network	p10
provider:segmentation_id	
router:external	False
shared	True
status	ACTIVE
subnets	
tenant_id	2e05722637b64192aede070137aeab22

```
root@tein:/home/tein#
```

그림 22 OpenStack flat network 생성

- o flat 네트워크를 생성한 후 OpenStack 대쉬보드 상에서 새로 만들어진 p10 네트워크에 192.168.40.0/24 대역의 서브넷을 추가한다.

그림 23 flat network에 서브넷 생성

- o 생성한 flat 네트워크에 서브넷 추가가 된 이후에는 br20과 vRouter를 연결한다. vRouter에 네트워크를 추가한다. VyOS에 새로운 eth0 네트워크 인터페이스를 추가한다.

그림 24 vRouter에 새로운 네트워크 인터페이스 추가

그리고 vRouter 상에서 OpenStack의 flat network와 같은 서브넷 대역의 아이피 설정을 해준다.

```
$ configure
# set interfaces ethernet eth0 address 192.168.40.8/24
# commit
```

Interface	IP Address	S/L	Description
-----	-----	---	-----
eth0	192.168.40.8/24	U/U	
eth1	210.114.90.80/8	U/U	
lo	127.0.0.1/8	U/U	
	::1/128		

그림 25 vRouter의 네트워크 인터페이스 설정 화면

이러한 설정들을 통해 GIST에 위치하고 있는 SANDBOX에 아래 그림26과 같은 환경을 구현할 수 있다.

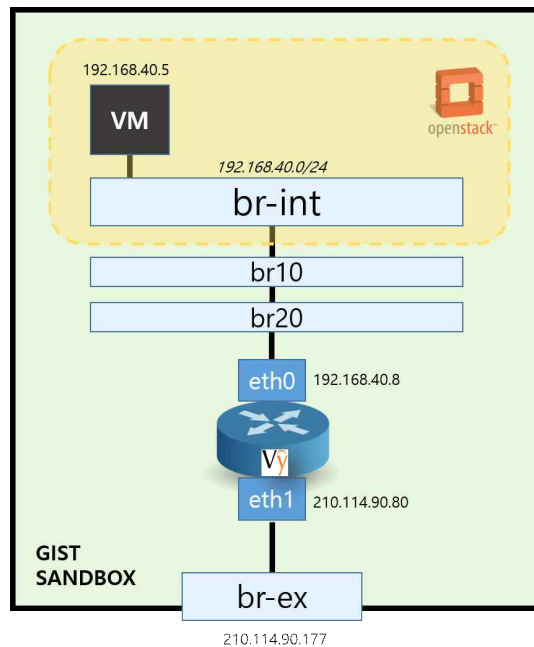


그림 26 SANDBOX 내부 OpenStack instance와 vRouter의 연결

- o OpenStack과 vRouter의 연결이 완료된 후에는 vRouter 상에서 OpenStack과 연결된 eth0에서 오는 패킷들을 eth1로 보내도록 설정한다.

```
$ configure
# set nat source rule 10 source address 192.168.40.8/24
# set nat source rule 10 outbound-interface eth0
# set nat source rule 10 translation address masquerade
```


4. End to End Security 검증

4.1. End to End Security 연결 검증

- o 상기와 같이 구성된 SmartX Box 기반 KOREN L3 연동의 실제 동작하는 단계를 제시함으로써 OpenStack Cloud의 SmartX 사이트 간 L3 연동을 검증한다.
- o 검증을 위한 시나리오는 다음과 같다. GIST에 위치한 SANDBOX 내부의 OpenStack instance 내부에서 SANDBOX 내부의 vRouter를 통해 POSTECH에 위치한 vRouter로 ping을 보냄으로써 vRouter를 통한 OpenStack instance의 L3 기반 연동을 확인할 수 있다.
- o 실험을 위해 그림 30과 같이 생성한 네트워크에 연결된 OpenStack instance를 생성한다. 이후 인스턴스 내부 콘솔에서 POSTECH의 SmartX Type C Box 내부의 vRouter에 ping을 보냄으로써 vRouter 기반의 L3 연동을 실험한다.



그림 27 OpenStack 네트워크 토폴로지

- o 아래 그림 31은 POSTECH에 위치한 vRouter의 cli 화면이다. GIST SANDBOX 내부의 OpenStack instance에서 보낸 ping을 tcpdump 명령어를 통해 확인하는 모습이다. ping이 vRouter에 도달하는 것으로 보아 vRouter가 L3 VPN tunnel을

통해 패킷을 제대로 전달하고 있음을 확인할 수 있다.

```
vyos@vyos# sudo tcpdump -i eth1 icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 65535 bytes
13:52:48.320596 IP 192.168.2.1 > all-systems.mcast.net: ICMP router advertisement lifetime 30:00 1: {192.168.2.1 0}, length 16
13:53:50.747651 IP 192.168.40.5 > 210.114.90.69: ICMP echo request, id 51713, seq 0, length 64
13:53:51.747363 IP 192.168.40.5 > 210.114.90.69: ICMP echo request, id 51713, seq 1, length 64
13:53:52.747611 IP 192.168.40.5 > 210.114.90.69: ICMP echo request, id 51713, seq 2, length 64
13:53:53.747748 IP 192.168.40.5 > 210.114.90.69: ICMP echo request, id 51713, seq 3, length 64
13:53:54.748001 IP 192.168.40.5 > 210.114.90.69: ICMP echo request, id 51713, seq 4, length 64
13:53:55.748165 IP 192.168.40.5 > 210.114.90.69: ICMP echo request, id 51713, seq 5, length 64
^C
7 packets captured
7 packets received by filter
0 packets dropped by kernel
```

그림 28 POSTECH SmartX Type C 내부 vRouter 콘솔

References

- [1] VyOS, http://vyos.net/wiki/Main_Page
- [2] Open vSwitch, <http://www.openvswitch.org>

오픈 SmartX 플랫폼을 위한 SmartX Box 기반 KOREN L3 연동

- 광주과학기술원의 확인과 허가 없이 이 문서를 무단 수정하여 배포하는 것을 금지합니다.
- 이 문서의 기술적인 내용은 프로젝트의 진행과 함께 별도의 예고 없이 변경될 수 있습니다.
- 본 문서와 관련된 대한 문의 사항은 아래의 정보를 참조하시길 바랍니다.
(Homepage: <https://nm.gist.ac.kr>, E-mail: ops@smartx.kr)

작성기관: 광주과학기술원

작성년월: 2015/12