

안승규의 블로그(Stay hungry, stay foolish)

- [지역로그](#)
- [태그로그](#)
- [미디어로그](#)
- [방명록](#)

DevStack Install - Ubuntu 16.04 LTS

[OpenStack](#) 2016.04.23 17:36 [신고](#)

[Server IP 정보]

eth0 : NAT type	(vmnet2)	192.168.75.138	Public Network
eth1 : Host-only type	(vmnet3)	192.168.230.138	Private Network

[Multi Node 의 경우 두번째 추가 Compute Node]

eth0 : NAT type	(vmnet2)	192.168.75.139	Public Network
eth1 : Host-only type	(vmnet3)	192.168.230.139	Private Network

[User 선택]

stack 유저로 생성

[visudo 세팅]

stack ALL=(ALL:ALL) NOPASSWD:ALL

[vi /etc/network/interfaces]

```
auto lo
iface lo inet loopback
```

```
auto ens33
iface ens33 inet static
    address 192.168.75.138
    netmask 255.255.255.0
    gateway 192.168.75.2
    dns-nameservers 8.8.8.8 8.8.4.4
```

```
auto ens34
iface ens34 inet static
    address 192.168.230.138
    netmask 255.255.255.0
```

[Host 변경]

```
mkdir -p ~/Documents/scripts
cd ~/Documents/scripts
```

```
vi servers.txt
192.168.230.138 devstack01
192.168.230.139 devstack02
```

```
vi 01-hosts-setup.sh
#!/bin/bash
```

```
SERVERLIST=$HOME/Documents/scripts/servers.txt
MASTER_IP="192.168.230.138"
MASTER_HOSTNAME="devstack01"
SSH_USER="stack"
```

```
function set_sshkey() {
    local server=$1
    if [[ $server == "$MASTER_IP" ]]; then
        if [[ ! -f "${HOME}/.ssh/id_rsa" ]]; then
            yes "" | ssh-keygen -t rsa -N ""
        else
            echo "skip to create ssh-keygen"
        fi
    fi
    cat ~/.ssh/id_rsa.pub | ssh $SSH_USER@$server -oStrictHostKeyChecking=no \
        "if [ ! -f ~/.ssh/authorized_keys ] || ! grep -q ${MASTER_HOSTNAME} ~/.ssh/authorized_keys; then \
            umask 077; test -d .ssh || mkdir -p .ssh; cat >> ~/.ssh/authorized_keys; \
            fi"
    echo "$server ssh-key ..... done"
}
```

```
function change_hostname() {
    local server=$1
    local hostname=$2
    echo ${hostname} | ssh $SSH_USER@$server \
        "if ! grep -q ${hostname} /etc/hostname; then \
            sudo su -c 'cat > /etc/hostname'; \
            sudo hostname -F /etc/hostname; \
            fi"
    echo "$server $hostname ..... done"
}
```

```
function change_hostfile() {
    local server=$1
    cat servers.txt | ssh $SSH_USER@$server \
        "if ! grep -q ${MASTER_HOSTNAME} /etc/hosts; then \
            sudo su -c 'cat >> /etc/hosts'; \
            fi"
    echo "$server hostfile .... done"
}
```

```
echo "setting sshkey ....."
while read line; do
    if [[ $(echo $line | cut -c1) != "#" ]]; then
        server=$(echo $line | awk '{print $1}')
        set_sshkey $server
    fi
done < $SERVERLIST
```

```
echo "changing hostname ....."
while read line; do
    if [[ $(echo $line | cut -c1) != "#" ]]; then
        server=$(echo $line | awk '{print $1}')
        hostname=$(echo $line | awk '{print $2}')
        change_hostname $server $hostname
    fi
done
```

```

fi
done < $SERVERLIST

echo "changing hosts file ....."
while read line; do
    if [[ $(echo $line | cut -c1) != "#" ]]; then
        server=$(echo $line | awk '{print $1}')
        change_hostfile $server
    fi
done < $SERVERLIST

```

[NTP 세팅]

```

vi 02-ntp-setup.sh
#!/bin/bash

```

```

SERVERLIST=$HOME/Documents/scripts/servers.txt
MASTER_IP="192.168.230.138"
SSH_USER="stack"

```

```

function ntp_master_setup() {
    local server=$1
    echo $server | ssh ${SSH_USER}@$server \
    "sudo apt-get update; \
    sudo apt-get install -y bridge-utils libvirt-bin ntp ntpdate; \
    if ! grep -q 'server 127.127.1.0' /etc/ntp.conf; then \
        sudo sed -i 's/server 0.ubuntu.pool.ntp.org/#server 0.ubuntu.pool.ntp.org/g' /etc/ntp.conf; \
        sudo sed -i 's/server 1.ubuntu.pool.ntp.org/#server 1.ubuntu.pool.ntp.org/g' /etc/ntp.conf; \
        sudo sed -i 's/server 2.ubuntu.pool.ntp.org/#server 2.ubuntu.pool.ntp.org/g' /etc/ntp.conf; \
        sudo sed -i 's/server 3.ubuntu.pool.ntp.org/server time.bora.net/g' /etc/ntp.conf; \
        sudo sed -i 's/server ntp.ubuntu.com/server 127.127.1.0/g' /etc/ntp.conf; \
        sudo sed -i 's/restrict 127.0.0.1/restrict 192.168.0.0 mask 255.255.0.0 nomodify notrap/g' /etc/ntp.conf; \
        sudo service ntp restart; \
    fi; \
    sudo ntpdate -u time.bora.net; \
    sudo virsh net-destroy default; \
    sudo virsh net-undefine default"
}

```

```

function ntp_slave_setup() {
    local server=$1
    echo $server | ssh ${SSH_USER}@$server \
    "sudo apt-get update; \
    sudo apt-get install -y bridge-utils libvirt-bin ntp ntpdate; \
    if ! grep -c ${MASTER_IP} /etc/ntp.conf; then \
        sudo sed -i 's/server 0.ubuntu.pool.ntp.org/#server 0.ubuntu.pool.ntp.org/g' /etc/ntp.conf; \
        sudo sed -i 's/server 1.ubuntu.pool.ntp.org/#server 1.ubuntu.pool.ntp.org/g' /etc/ntp.conf; \
        sudo sed -i 's/server 2.ubuntu.pool.ntp.org/#server 2.ubuntu.pool.ntp.org/g' /etc/ntp.conf; \
        sudo sed -i 's/server 3.ubuntu.pool.ntp.org/#server 3.ubuntu.pool.ntp.org/g' /etc/ntp.conf; \
        sudo sed -i 's/server ntp.ubuntu.com/server ${MASTER_IP}/g' /etc/ntp.conf; \
        sudo service ntp restart; \
    fi; \
    sudo ntpdate -u ${MASTER_IP}; \
    sudo virsh net-destroy default; \
    sudo virsh net-undefine default"
}

```

```
}
```

```
echo "ntp master setting ....."
while read line; do
    if [[ $(echo $line | cut -c1) != "#" ]]; then
        server=$(echo $line | awk '{print $1}')
        if [[ $server == "$MASTER_IP" ]]; then
            ntp_master_setup $server
        fi
    fi
done < $SERVERLIST

echo "ntp slave setting ....."
while read line; do
    if [[ $(echo $line | cut -c1) != "#" ]]; then
        server=$(echo $line | awk '{print $1}')
        if [[ $server != "$MASTER_IP" ]]; then
            ntp_slave_setup $server
        fi
    fi
done < $SERVERLIST
```

[local.conf 파일]

```
mkdir -p ~/Documents/github
cd github
git clone https://github.com/openstack-dev/devstack.git
cd devstack
```

vi local.conf

```
[[local|localrc]]
HOST_IP=192.168.75.138
SERVICE_HOST=192.168.75.138
MYSQL_HOST=192.168.75.138
RABBIT_HOST=192.168.75.138
GLANCE_HOSTPORT=192.168.75.138:9292
ADMIN_PASSWORD=secret
DATABASE_PASSWORD=secret
RABBIT_PASSWORD=secret
SERVICE_PASSWORD=secret
```

```
# Do not use Nova-Network
disable_service n-net
```

```
# Neutron service
enable_service neutron
enable_service q-svc
enable_service q-agt
enable_service q-dhcp
enable_service q-l3
enable_service q-meta
```

```
# Neutron options
Q_USE_SECGROUP=True
FLOATING_RANGE="192.168.75.0/24"
```

```
FIXED_RANGE="10.0.0.0/24"
Q_FLOATING_ALLOCATION_POOL=start=192.168.75.193,end=192.168.75.254
PUBLIC_NETWORK_GATEWAY="192.168.75.2"
Q_L3_ENABLED=True
PUBLIC_INTERFACE=ens33

# Open vSwitch provider networking configuration
Q_USE_PROVIDERNET_FOR_PUBLIC=True
OVS_PHYSICAL_BRIDGE=br-ex
PUBLIC_BRIDGE=br-ex
OVS_BRIDGE_MAPPINGS=public:br-ex

# Nova service
enable_service n-api
enable_service n-cpu
enable_service n-cond
enable_service n-sch
enable_service n-novnc
enable_service n-cauth

# Cinder service
enable_service cinder
enable_service c-api
enable_service c-vol
enable_service c-sch
enable_service c-bak

# Tempest service
enable_service tempest

# Swift service
enable_service s-proxy
enable_service s-object
enable_service s-container
enable_service s-account

# Heat service
enable_service heat
enable_service h-api
enable_service h-api-cfn
enable_service h-api-cw
enable_service h-eng

# Enable plugin neutron-lbaas, octavia
enable_plugin neutron-lbaas https://git.openstack.org/openstack/neutron-lbaas master
enable_plugin octavia https://git.openstack.org/openstack/octavia

# Enable plugin Magnum
#enable_plugin magnum https://github.com/openstack/magnum master
#enable_plugin magnum-ui https://github.com/openstack/magnum-ui master

# Enable plugin Monasca (Ubuntu 16.04 사용 시 Systemctl 에 맞게 수정 필요)
enable_plugin monasca-api https://github.com/openstack/monasca-api master
enable_plugin monasca-log-api https://github.com/openstack/monasca-log-api master
MONASCA_API_IMPLEMENTATION_LANG=\
${MONASCA_API_IMPLEMENTATION_LANG:-python}
```

```
MONASCA_PERSISTER_IMPLEMENTATION_LANG=\
${MONASCA_PERSISTER_IMPLEMENTATION_LANG:-python}
MONASCA_METRICS_DB=${MONASCA_METRICS_DB:-influxdb}
```

```
# Cinder configuration
VOLUME_GROUP="cinder-volumes"
VOLUME_NAME_PREFIX="volume-"
```

```
# Images
# Use this image when creating test instances
IMAGE_URLS+="http://download.cirros-cloud.net/0.3.4/cirros-0.3.4-x86_64-disk.img"
# Use this image when working with Orchestration (Heat)
IMAGE_URLS+="https://download.fedoraproject.org/pub/fedora/linux/releases/23/Cloud/x86_64/Images/Fedora-Cloud-Base-23-20151030.x86_64.qcow2"
```

```
KEYSTONE_CATALOG_BACKEND=sql
API_RATE_LIMIT=False
SWIFT_HASH=testing
SWIFT_REPLICAS=1
VOLUME_BACKING_FILE_SIZE=70000M
```

```
LOGFILE=$DEST/logs/stack.sh.log
```

```
# A clean install every time
RECLONE=yes
```

[Compute Node 추가]

vi local.conf

```
[[local|localrc]]
HOST_IP=192.168.75.139
SERVICE_HOST=192.168.75.138
MYSQL_HOST=192.168.75.138
RABBIT_HOST=192.168.75.138
GLANCE_HOSTPORT=192.168.75.138:9292
ADMIN_PASSWORD=secret
MYSQL_PASSWORD=secret
RABBIT_PASSWORD=secret
SERVICE_PASSWORD=secret
```

```
# Neutron options
PUBLIC_INTERFACE=ens33
ENABLED_SERVICES=n-cpu,n-novnc,rabbit,q-agt
```

```
LOGFILE=$DEST/logs/stack.sh.log
```

[설치 실행]

```
./stack.sh
```

[스토리지 마운트]

```
sudo mount -t xfs -o loop,noatime,nodiratime,nobarrier,logbufs=8 /opt/stack/data/swift/drives/images/swift.img  
/opt/stack/data/swift/drives/sdb1
```

```
sudo losetup /dev/loop1 /opt/stack/data/cinder-volumes-default-backing-file
```

```
sudo losetup /dev/loop2 /opt/stack/data/cinder-volumes-lvmdriver-1-backing-file
```

[CPU, Ram, Disk Overcommit 세팅]

```
vi /etc/nova/nova.conf
```

```
scheduler_default_filters = ..., CoreFilter      # CoreFilter 추가  
cpu_allocation_ratio=50.0  
ram_allocation_ratio=16.0  
disk_allocation_ratio=50.0
```

[서비스 실행]

```
screen -c stack-screenrc
```

[VM 생성]

```
. openrc admin demo
```

```
openstack project list  
openstack security group list
```

default sec group rule 추가

```
openstack security group rule create --proto icmp --src-ip 0.0.0.0/0 --dst-port -1 --ingress 2d95031b-132b-4d46-  
aacd-f392cdd8c4fb
```

```
openstack security group rule create --proto tcp --src-ip 0.0.0.0/0 --dst-port 1:65535 --ingress 2d95031b-132b-  
4d46-aacd-f392cdd8c4fb
```

private key 생성

```
openstack keypair create --public-key ~/.ssh/id_rsa.pub magnum-key
```

```
openstack flavor list  
openstack image list  
openstack network list
```

nova boot

```
openstack server create --image 7e688989-e59b-4b20-a562-1de946ee91e9 --flavor m1.tiny --nic net-  
id=f57b8f2c-cd67-4d49-b38c-393dbb773c9b --key-name magnum-key --security-group default test-01
```

floating ip 생성 및 서버 할당

```
openstack ip floating create public  
openstack ip floating list  
openstack ip floating add 192.168.75.194 test-01
```

```
# Router 보기|  
sudo ip netns
```

```
qdhcp-f57b8f2c-cd67-4d49-b38c-393dbb773c9b
qrouter-b46e14d5-4ef5-4bfa-8dc3-463a982688ab
```

[tcpdump 방법]

Compute Node

```
[vm] -> tap:[qbrb97b5aa3-f8 Linux Bridge]:qvbb97b5aa3-f8 -> qvob97b5aa3-f8:[OVS br-int Bridge]:patch-tun -
> patch-int:[OVS br-tun Bridge]:br-tun ->
```

Network Node

```
br-tun:OVS br-tun Bridge:patch-int -> patch-tun:OVS br-int Bridge:qr-c163af1e-53 ->
qr-c163af1e-53:qrouter(Namespace) -> qg-d8187261-68:qg(Namespace) ->
qg-d8187261-68:OVS br-int Bridge:int-br-ex -> phy-br-ex:OVS br-ex Bridge -> NIC
```

```
sudo tcpdump -n -e -i qbrb97b5aa3-f8 | grep 10.0.0.3
sudo tcpdump -n -e -i qvbb97b5aa3-f8 | grep 10.0.0.3
sudo tcpdump -n -e -i qvob97b5aa3-f8 | grep 10.0.0.3
sudo ip netns exec qrouter-b46e14d5-4ef5-4bfa-8dc3-463a982688ab tcpdump -n -e -i qr-c163af1e-53 | grep
10.0.0.3
```

[Magnum k8s 생성]

```
cd ~/Documents/github/devstack/files
wget https://fedorapeople.org/groups/magnum/fedora-21-atomic-5.qcow2
glance image-create --name fedora-21-atomic-5 \
    --visibility public \
    --disk-format qcow2 \
    --os-distro fedora-atomic \
    --container-format bare < fedora-21-atomic-5.qcow2
```

```
magnum service-list
```

```
magnum baymodel-create --name k8sbaymodel \
    --image-id fedora-21-atomic-5 \
    --keypair-id magnum-key \
    --external-network-id public \
    --dns-nameserver 8.8.8.8 \
    --flavor-id m1.small \
    --docker-volume-size 5 \
    --network-driver flannel \
    --coe kubernetes
```

```
magnum baymodel-list
```

```
magnum bay-create --name k8sbay --baymodel k8sbaymodel --node-count 1
```

```
neutron lb-pool-list
```

```
neutron lb-vip-list
```

```
neutron lb-member-list
```

```
magnum bay-list
```

[magnum 클러스터 생성 에러 시 수동으로 할 때 삭제해야 할 것]

```
# floating ip 삭제 - api-pool-vip, kube-master, kube-node
```



```
openstack ip floating list  
sudo ip netns exec qrouter-2f49aeb4-421c-4994-923a-5aafe453fa3d ip a
```

```
# api.pool.vip 삭제  
neutron lb-vip-list  
neutron lb-pool-list  
neutron lb-member-list
```

```
# private network 삭제  
openstack network list
```

```
# router 삭제, external gateway 삭제  
openstack router list  
openstack port list  
openstack router remove port      (gateway 를 제거)  
openstack router remove subnet    (subnet 을 제거)
```



Thích Hãy là người đầu tiên trong số bạn bè của bạn
thích nội dung này.

Posted by OpenStack Korea 부대표 Stephen Ahn

[트랙백 0개](#), [댓글 2개](#)가 달렸습니다

[이전](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) ... [188](#) [다음](#)



by **Stephen Ahn**

- [관리자](#)
- [글쓰기](#)

카테고리

- T** 분류 전체보기 (188)
 - 여행 (10)
 - 중국 (7)
 - 사진>>포토샵 (1)
 - Miscellaneous (11)
 - IT 이야기 (6)
 - 프로그래밍 (63)
 - 영어 (12)
 - Flex (3)
 - Linux (12)
 - Mac (10)
 - MySQL (4)
 - OpenStack (33)
 - Cloud Computing (2)
 - PostgreSQL (1)
 - Algorithm (2)
 - Container (7)

태그목록

- [클래스](#)
- [모임](#)
- [Mac](#)
- [미녀삼총사](#)
- [php](#)
- [git](#)
- [회사](#)
- [hbase](#)
- [객체](#)
- [OpenStack](#)
- [중국](#)
- [뉴욕](#)
- [환경세팅](#)
- [iPhone](#)
- [install](#)
- [ubuntu](#)
- [Javascript](#)
- [mysql](#)
- [iptables](#)
- [포토샵](#)
- [하늘](#)
- [flex](#)
- [Hadoop](#)
- [nova](#)
- [기다림](#)
- [데이터타입](#)
- [Python](#)
- [Folsom Design Summit](#)
- [개발자](#)
- [RPC](#)

최근에 달린 댓글

- [최신 마스터입니다..](#) OpenStack Korea 부대표 Stephen Ahn 06.10
- [오픈스택 버전이 어...](#) 회사원 05.28
- [퍼가겠습니다. 분기...](#) 박종승 01.22
- [좋은자료 퍼가겠습...](#) chobo 2015
- [Vagrant VM 접속하...](#) 사진우주 2015

글 보관함

- [2016/05](#) (2)
- [2016/04](#) (3)
- [2016/01](#) (3)
- [2015/12](#) (2)
- [2015/08](#) (4)

링크

- [태우의 블로그.](#)
- [에이콘 출판사.](#)



[지역로그](#) : [태그로그](#) : [미디어로그](#) : [방명록](#) : [관리자](#) : [글쓰기](#)
[Stephen Ahn's](#) Blog is powered by [Daum](#) / Designed by [Tistory](#)