목차

네트워킹 옵션 2 - 셀프 서비스 네트워크 3
구성 요소 설치 3
서버 구성 요소 구성 3
모듈러 레이어 2 (ML2) 플러그인 구성 5
Linux bridge agent 구성 6
Layer-3 에이전트 구성 7
DHCP 에이전트 구성 8

네트워킹 옵션 2 - 셀프 서비스 네트워크

controller 노드 에 네트워킹 구성 요소를 설치하고 구성 합니다.

구성 요소 설치

```
[root@controller ~]# yum install openstack-neutron openstack-neutron-ml2 \
  openstack-neutron-linuxbridge ebtables
```

서버 구성 요소 구성

1. /etc/neutron/neutron.conf 파일을 편집하고 다음 조치를 완료하십시오.

- [database]섹션에서 데이터베이스 액세스 구성:

```
[database]
# ...
connection = mysql+pymysql://neutron:NEUTRON_DBPASS@controller/neutron
```

NEUTRON_DBPASS 부분을 이전에 설정한 암호로 바꿉니다 .

// [database]섹션에서 connection을 제외한 나머지 다른 옵션을 주석 처리하거나 제거 하십시오.//

- [DEFAULT]섹션에서 모듈러 레이어 2 (ML2) 플러그인, 라우터 서비스 및 IP 주소 오버래핑 사용 설정:

```
[DEFAULT]
# ...
core_plugin = ml2
service_plugins = router
allow_overlapping_ips = true
```

- [DEFAULT]섹션에서 RabbitMQ 메시지 큐 액세스 구성:

```
[DEFAULT]
# ...
transport_url = rabbit://openstack:RABBIT_PASS@controller
```

RABBIT_PASS부분을 이전에 설정한 암호로 바꿉니다.

- [DEFAULT]와 [keystone_authtoken]섹션에서 아이덴티티 서비스 액세스 구성:

```
[DEFAULT]
# ...
auth_strategy = keystone

[keystone_authtoken]
# ...
www_authenticate_uri = http://controller:5000
auth_url = http://controller:5000
memcached_servers = controller:11211
auth_type = password
project_domain_name = default
user_domain_name = default
project_name = service
username = neutron
password = NEUTRON_PASS
```

NEUTRON_PASS부분을 이전에 설정한 비밀번호로 바꿉니다. *[keystone_authtoken] 섹션 에서 나머지 다른 옵션을 주석 처리하거나 제거 하십시오.*

- [DEFAULT]와 [nova]섹션에서 compute 노드로의 네트워크 토폴로지 변경 알림 구성:

```
[DEFAULT]
# ...
notify_nova_on_port_status_changes = true
notify_nova_on_port_data_changes = true

[nova]
# ...
auth_url = http://controller:5000
auth_type = password
project_domain_name = default
user_domain_name = default
region_name = RegionOne
project_name = service
username = nova
password = NOVA_PASS
```

NOVA_PASS 부분을 이전에 설정한 비밀번호로 바꿉니다.

- [oslo_concurrency]섹션에서 잠금 경로를 구성 :

```
[oslo_concurrency]
# ...
lock_path = /var/lib/neutron/tmp
```

모듈러 레이어 2 (ML2) 플러그인 구성

ML2 플러그인은 Linux 브리지 메커니즘을 사용하여 인스턴스 용 레이어 2 (브리징 및 스위칭) 가상 네트워킹 인프라를 구축합니다.

1. `/etc/neutron/plugins/ml2/ml2_conf.ini`파일을 편집하고 다음 조치를 완료하십시오.

- `[ml2]`섹션에서 FLAT, VLAN, 그리고 VXLAN 네트워크를 활성화합니다

```
[ml2]
# ...
type_drivers = flat,vlan,vxlan
```

- `[ml2]`섹션에서 VXLAN 셀프 서비스 네트워크를 활성화합니다

```
[ml2]
# ...
tenant_network_types = vxlan
```

- `[ml2]`섹션에서 Linux bridge와 L2-population 메커니즘을 활성화 :

```
[ml2]
# ...
mechanism_drivers = linuxbridge,l2population
```

ML2 플러그인을 구성한 후 `type_drivers`옵션 에서 값을 제거 하면 데이터베이스 불일치가 발생할 수 있습니다. `Linux bridge agent`는 VXLAN 오버레이 네트워크 만 지원합니다.

- `[ml2]`섹션에서 포트 보안 확장 드라이버를 사용 :

```
[ml2]
# ...
extension_drivers = port_security
```

- `[ml2_type_flat]`섹션에서 FLAT 네트워크로 provider 가상 네트워크를 구성 :

```
[ml2_type_flat]
# ...
flat_networks = provider
```

- `[ml2_type_vxlan]`섹션에서 셀프 서비스 네트워크를 위한 VXLAN 네트워크 식별자의 범위를 구성합니다

```
[ml2_type_vxlan]
# ...
vni_ranges = 1:1000
```

- [securitygroup]섹션에서 보안 그룹 규칙의 효율성을 높이기 위해 ipset 활성화 :

```
[securitygroup]
# ...
enable_ipset = true
```

Linux bridge agent 구성

Linux bridge agent는 인스턴스 용 레이어 2 (브리징 및 스위칭) 가상 네트워킹 인프라를 구축하고 보안 그룹을 처리합니다.

1. /etc/neutron/plugins/ml2/linuxbridge_agent.ini파일을 편집하고 다음 조치를 완료하십시오.

- [linux_bridge]섹션에서 제공 물리적 네트워크 인터페이스로 제공 가상 네트워크를 매핑 :

```
[linux_bridge]
physical_interface_mappings = provider:PROVIDER_INTERFACE_NAME
```

PROVIDER_INTERFACE_NAME부분을 기본 Provider 물리 네트워크 인터페이스의 이름으로 바꿉니다.

- [vxlan] 섹션에서 VXLAN 오버레이 네트워크를 활성화 하고, 오버레이 네트워크를 처리하는 물리적 네트워크 인터페이스의 IP 주소를 구성하고, L2-population를 활성화합니다.

```
[vxlan]
enable_vxlan = true
local_ip = OVERLAY_INTERFACE_IP_ADDRESS
l2_population = true
```

OVERLAY_INTERFACE_IP_ADDRESS부분을 오버레이 네트워크를 처리하는 기본 물리적 네트워크 인터페이스의 IP 주소로 바꿉니다. 예제 아키텍처는 관리 인터페이스를 사용하여 트래픽을 다른 노드로 터널링합니다. 따라서 OVERLAY_INTERFACE_IP_ADDRESS를 컨트롤러 노드의 관리 IP 주소로 교체 하십시오.

- [securitygroup]섹션에서 보안 그룹을 활성화하고 리눅스 브리지 iptables 방화벽 드라이버를 구성 :

```
[securitygroup]
# ...
enable_security_group = true
firewall_driver =
neutron.agent.linux.iptables_firewall.IptablesFirewallDriver
```

- 다음 sysctl값이 모두 1로 설정되어 있는지 확인하여 Linux 운영 체제 커널이 네트워크 브리지 필터를 지원하는지 확인합니다.

```
net.bridge.bridge-nf-call-iptables
net.bridge.bridge-nf-call-ip6tables
```

네트워킹 브리지 지원을 활성화하려면 일반적으로 br_netfilter 커널 모듈을 로드해야합니다. 만약 sysctl 에서 해당 Kernel parameter 목록이 안보인 경우 아래와 같이 확인한다.

```
[root@controller ~]# lsmod | grep "br_netfilter"
br_netfilter           22256  0
bridge                151336  1 br_netfilter
```

출력이 안된다면 모듈이 탑재되지 않은것이므로 다음과 같이 모듈을 탑재 한다.

```
[root@controller ~]# modprobe br_netfilter
```

영구적으로 적용하려면 /etc/modules-load.d/br_netfilter.conf파일을 만들어 준다.

```
br_netfilter
```

Layer-3 에이전트 구성

Layer-3 (L3) 에이전트는 셀프 서비스 가상 네트워크에 대한 라우팅 및 NAT 서비스를 제공합니다.

1. /etc/neutron/l3_agent.ini파일을 편집하고 다음 조치를 완료하십시오.

- [DEFAULT]섹션에서 리눅스 브리지 인터페이스 드라이버를 구성 :

```
[DEFAULT]
# ...
interface_driver = linuxbridge
```

DHCP 에이전트 구성

DHCP 에이전트는 가상 네트워크에 DHCP 서비스를 제공합니다.

1. `/etc/neutron/dhcp_agent.ini`파일을 편집하고 다음 조치를 완료하십시오.

   - `[DEFAULT]`섹션에서 리눅스 브리지 인터페이스 드라이버, `dnsmasq` DHCP 드라이버를 구성하고, Provider 네트워크에 인스턴스가 네트워크를 통해 메타 데이터를 액세스 할 수 있도록 메타 데이터를 분리구성합니다:

```
[DEFAULT]
# ...
interface_driver = linuxbridge
dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq
enable_isolated_metadata = true
```

From:
https://atl.kr/dokuwiki/ - **AllThatLinux!**

Permanent link:
https://atl.kr/dokuwiki/doku.php/%EB%84%A4%ED%8A%B8%EC%9B%8C%ED%82%B9_%EC%98%B5%EC%85%98_2_-_%EC%85%80%ED%94%84_%EC%84%9C%EB%B9%84%EC%8A%A4_%EB%84%A4%ED%8A%B8%EC%9B%8C%ED%81%AC

Last update: **2020/09/24 02:17**