

目 录

1	设置网络	2
2	配置 ironic 网络	3
3	启动 ironic	3
4	创建物理机需要的镜像	4
5	创建物理机需要的 flavor	4
6	重启 nova 服务	5
7	配置 tftpd	5
8	配置 iptable	5

1 设置网络

设置网络的思路：

1. 创建一个外部网络 public 和一个内部网络 private。
2. 创建内部网络子网池。
3. 创建内部子网和外部子网。
4. 创建一个路由器，将内部子网连接到路由器，再将外部网络设置为路由器的外部网关。

```

1 # 创建 private 网络
2 openstack --os-cloud devstack-admin --os-region RegionOne network create --
   project 6eaff7ecc1694fcd94532bad5f09e17f private
3 # 创建 public 网络
4 openstack --os-cloud devstack-admin --os-region RegionOne network create public
   --external --default --provider-network-type flat --provider-physical-
   network public
5 # 创建子网池
6 openstack --os-cloud devstack-admin --os-region RegionOne subnet pool create
   shared-default-subnetpool --default-prefix-length 26 --pool-prefix 10.0.0.0
   /22 --share --default
7 openstack --os-cloud devstack-admin --os-region RegionOne subnet pool create
   shared-default-subnetpool --default-prefix-length 64 --pool-prefix fd99
   :0295:1537::/56 --share --default
8 # 创建 private 子网
9 openstack --os-cloud devstack-admin --os-region RegionOne subnet create --
   project 6eaff7ecc1694fcd94532bad5f09e17f --ip-version 4 --gateway 10.1.0.1
   --subnet-pool 700abf50-bd39-4e09-bbab-780d21cdd9b --network 056b7a85-726a
   -4d18-abef-59f2a43aa16d private-subnet
10 openstack --os-cloud devstack-admin --os-region RegionOne subnet create --
   project 6eaff7ecc1694fcd94532bad5f09e17f --ip-version 6 --subnet-pool 254
   e2198-20c0-46d1-830a-ddfeaa14d3f1 --ipv6-ra-mode slaac --ipv6-address-mode
   slaac --network 056b7a85-726a-4d18-abef-59f2a43aa16d ipv6-private-subnet
11 # 创建 public 子网
12 openstack --os-cloud devstack-admin --os-region RegionOne subnet create --ip-
   version 4 --network 6b28596f-738f-4a0d-9864-0183801cdd49 --subnet-range 172
   .24.4.0/24 --no-dhcp public-subnet
13 openstack --os-cloud devstack-admin --os-region RegionOne subnet create --ip-
   version 6 --gateway 2001:db8::2 --network 6b28596f-738f-4a0d-9864-0183801
   cdd49 --subnet-range 2001:db8::/64 --no-dhcp ipv6-public-subnet
14 # 创建路由器
15 openstack --os-cloud devstack-admin --os-region RegionOne router create --
   project 6eaff7ecc1694fcd94532bad5f09e17f router1
16 # 将 private subnet 加入 router
17 openstack --os-cloud devstack-admin --os-region RegionOne router add subnet
   ff71a343-a4fb-4a6a-b7fc-f013566b31cc 26e240db-8cc9-4938-831a-5710d689426a
18 openstack --os-cloud devstack-admin --os-region RegionOne router add subnet
   ff71a343-a4fb-4a6a-b7fc-f013566b31cc ebd2af8f-3450-49f4-abff-2d4577bc9b40
19 # 设置外部网关，6b28596f-738f-4a0d-9864-0183801cdd49 是 public 网络的 uuid

```

```

20 openstack --os-cloud devstack-admin --os-region RegionOne router set --external
    --gateway 6b28596f-738f-4a0d-9864-0183801cdd49 ff71a343-a4fb-4a6a-b7fc-
    f013566b31cc

```

2 配置 ironic 网络

```

1 # 获得内网的uuid
2 nova network-list | grep private | cut -d"|" -f2 | cut -d" " -f2
3
4 # 056b7a85-726a-4d18-abef-59f2a43aa16d是内网的uuid
5 neutron port-create 056b7a85-726a-4d18-abef-59f2a43aa16d
6 sudo ip netns exec qdhcp-056b7a85-726a-4d18-abef-59f2a43aa16d ip link list
7 sudo ovs-vsctl get port tap20f9e951-ec tag
8 sudo ip link show ovs-tap
9 sudo ip link show brbm-tap
10 sudo ip link add brbm-tap type veth peer name ovs-tap
11 sudo ip link set dev brbm-tap up
12 sudo ip link set dev ovs-tap up
13 sudo ovs-vsctl --if-exists del-port ovs-tap -- add-port br-int ovs-tap tag=1
14 sudo ovs-vsctl --if-exists del-port brbm-tap -- add-port brbm brbm-tap
15 openstack port delete f44258e5-8805-460e-9de7-a7cc1af501a8
16
17 # 建立网桥
18 # 将内网设置为可共享的
19 openstack network set 056b7a85-726a-4d18-abef-59f2a43aa16d --share
20
21 # ff71a343-a4fb-4a6a-b7fc-f013566b31cc是router1的uuid
22 openstack router show router1 -f value -c id
23 # 获得网关地址
24 sudo ip netns exec qrouter-ff71a343-a4fb-4a6a-b7fc-f013566b31cc ip -4 route get
    8.8.8.8
25 ip route replace 10.0.0.0/22 via 172.24.4.5
26
27 # configure_ironic_networks
28 iniset /etc/ironic/ironic.conf neutron cleaning_network private

```

3 启动 ironic

```

1 # 启动 ironic
2 # run_process ir-api '/usr/local/bin/ironic-api --config-file=/etc/ironic/
    ironic.conf'
3 screen -S stack -X screen -t ir-api
4 screen -S stack -p ir-api -X logfile /home/pengsida/temp/logs/ir-api.log.2017
    -03-30-172117
5 screen -S stack -p ir-api -X log on
6 touch /home/pengsida/temp/logs/ir-api.log.2017-03-30-172117
7 bash -c 'cd '\''/home/pengsida/temp/logs'\'' && ln -sf '\''ir-api.log.2017
    -03-30-172117'\'' ir-api.log'

```

```

8  screen -S stack -p ir-api -X stuff '/usr/local/bin/ironic-api --config-file=/
   etc/ironic/ironic.conf & echo $! >/opt/stack/status/stack/ir-api.pid; fg ||
   echo "ir-api failed to start. Exit code: $?" | tee "/opt/stack/status/
   stack/ir-api.failure "^M'
9
10 # run_process ir-cond '/usr/local/bin/ironic-conductor --config-file=/etc/
   ironic/ironic.conf'
11 screen -S stack -X screen -t ir-cond
12 screen -S stack -p ir-cond -X logfile /home/pengsida/temp/logs/ir-cond.log.2017
   -03-30-172117
13 screen -S stack -p ir-cond -X log on
14 touch /home/pengsida/temp/logs/ir-cond.log.2017-03-30-172117
15 bash -c 'cd '\''/home/pengsida/temp/logs'\'' && ln -sf '\''ir-cond.log.2017
   -03-30-172117'\'' ir-cond.log'
16 screen -S stack -p ir-cond -X stuff '/usr/local/bin/ironic-conductor --config-
   file=/etc/ironic/ironic.conf & echo $! >/opt/stack/status/stack/ir-cond.pid
   ; fg || echo "ir-cond failed to start. Exit code: $?" | tee "/opt/stack/
   status/stack/ir-cond.failure "^M'
17
18 # restart_apache_server
19 sudo service apache2 stop
20 sudo service apache2 start
21
22 # configure_ironic_ssh_keypair
23 mkdir -p /opt/stack/data/ironic/ssh_keys
24 ssh-keygen -q -t rsa -P '' -f /opt/stack/data/ironic/ssh_keys/ironic_key
25 cat /opt/stack/data/ironic/ssh_keys/ironic_key.pub
26 sort -u -o /home/pengsida/.ssh/authorized_keys /home/pengsida/.ssh/
   authorized_keys
27 ssh -p 22 -o BatchMode=yes -o ConnectTimeout=15 -o StrictHostKeyChecking=no -i
   /opt/stack/data/ironic/ssh_keys/ironic_key pengsida@10.250.1.3 exit

```

4 创建物理机需要的镜像

```

1  # upload_baremetal_ironic_deploy
2  openstack image create ir-deploy-agent_ipmitool.kernel --public --disk-format=
   aki --container-format=aki
3  openstack image create ir-deploy-agent_ipmitool.initramfs --public --disk-
   format=ari --container-format=ari

```

5 创建物理机需要的 flavor

```

1  openstack flavor create --ephemeral 0 --ram 1280 --disk 10 --vcpus 1 baremetal
2  openstack flavor set baremetal --property cpu_arch=x86_64

```

6 重启 nova 服务

```

1  # stop_nova_compute
2  # stop_process n-cpu
3  cat /opt/stack/status/stack/n-cpu.pid | pkill -g
4  screen -S stack -p n-cpu -X kill
5
6  # start_nova_compute
7  # run_process n-cpu '/usr/local/bin/nova-compute --config-file /etc/nova/
   nova.conf'
8  screen -S stack -X screen -t n-cpu
9  screen -S stack -p n-cpu -X logfile /home/pengsida/temp/logs/n-cpu.log.2017
   -03-30-172117
10 screen -S stack -p n-cpu -X log on
11 touch /home/pengsida/temp/logs/n-cpu.log.2017 -03-30-172117
12 bash -c 'cd `ls /home/pengsida/temp/logs ` && ln -sf `ls `n-cpu.log.2017
   -03-30-172117` n-cpu.log'
13 screen -S stack -p n-cpu -X stuff '/usr/local/bin/nova-compute --config-file /
   etc/nova/nova.conf & echo $! >/opt/stack/status/stack/n-cpu.pid; fg || echo
   "n-cpu failed to start. Exit code: $?" | tee "/opt/stack/status/stack/n-
   cpu.failure"^M'

```

7 配置 tftpd

```

1  # configure_tftpd
2  sudo service tftpd-hpa stop
3  sudo tee /etc/init/tftpd-hpa.override
4  sudo cp /opt/stack/ironic/devstack/tools/ironic/templates/tftpd-xinetd.template
   /etc/xinetd.d/tftp
5  sudo sed -e 's|TFTPBOOT_DIR|/opt/stack/data/ironic/tftpboot|g' -i /etc/
   xinetd.d/tftp
6  chmod -R 0755 /opt/stack/data/ironic/tftpboot
7
8  sudo service xinetd restart

```

8 配置 iptable

```

1  # configure_iptables
2  sudo modprobe nf_conntrack_tftp
3  sudo modprobe nf_nat_tftp
4  sudo iptables -I INPUT -p udp --dport 67:68 --sport 67:68 -j ACCEPT
5  sudo iptables -I INPUT -d 10.250.1.3 -p udp --dport 69 -j ACCEPT
6  sudo iptables -I INPUT -d 10.250.1.3 -p tcp --dport 6385 -j ACCEPT
7  sudo iptables -I INPUT -d 10.250.1.3 -p tcp --dport 8080 -j ACCEPT
8  sudo iptables -I INPUT -d 10.250.1.3 -p tcp --dport 9292 -j ACCEPT
9  sudo iptables -I INPUT -d 10.250.1.3 -p tcp --dport 3928 -j ACCEPT

```