Server as Code

RSS

Setting the Default MTU in Neutron VXLAN Networks to be 1500

Dealing with MTU issues is no fun. They are hard to diagnose. One of the issues I have commonly had is when I create a Docker node in a tenant VXLAN based Neutron network in an OpenStack cloud, and the interface in the virtual machine gets a MTU of 1450 (default 1500 - 50 for VXLAN) but Docker sets up an interface with an MTU of 1500. This will fail in all kinds of ugly ways that aren’t obvious, unless you know what MTU issues look like.

Below I show the docker0 interface on a docker node that is running in a default Neutron VXLAN. Note 1500 on the docker0 interface and 1450 on ens3.

```
$ ip ad sh docker0
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group defaul
    link/ether 02:42:90:79:85:26 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 scope global docker0
```

```
       valid_lft forever preferred_lft forever
    inet6 fe80::42:90ff:fe79:8526/64 scope link
       valid_lft forever preferred_lft forever
$ ip ad sh ens3
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1450 qdisc pfifo_fast state UP group defaul
    link/ether fa:16:3e:39:e5:c7 brd ff:ff:ff:ff:ff:ff
    inet 10.50.0.18/24 brd 10.50.0.255 scope global ens3
       valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fe39:e5c7/64 scope link
       valid_lft forever preferred_lft forever
```

So how do we change this?

1. Ensure the physical network has an MTU > 1550.
2. Make a few configuration changes to Neutron.

Physical network

Whichever interface (usually a bond) that holds the “underlay” tunnel IP address, ie. the one that Open vSwitch listens on, needs to have an MTU > 1550. Usually people set it to “jumbo frames”, ie. 9000 or higher.

Say you have some Juniper switches, the MTU of the bond interface can be set to have an MTU of 9200. We’d want to do that for all the interfaces that are part of the tunnel/underlay setup.

```
switch# set interfaces ae0 mtu 9200
# commit etc
switch> show interfaces ae0 | match MTU
  Link-level type: Ethernet, MTU: 9200, Speed: 2Gbps, BPDU Error: None, MAC-REWRITE Error:
```

Now that the physical infrastructure will do > 1550 we can configure Neutron.

Configure Neutron to Set the MTU of VXLANs to 1500

First, I should note this is an Otaka based cloud I’m working on.

I want a couple things:

- VXLAN interfaces in instances to have an MTU of 1500
- Provider networks to also have a MTU of 1500 so that they can communicate with the Internet

To do this I set three variables in Neutron configuration files:

1. neutron.conf: [DEFAULT] - global_physnet_mtu = 1550
2. ml2_conf.ini: [DEFAULT] - path_mtu = 1550

3. ml2_conf.ini: [DEFAULT] - physical_network_mtus = provider:1500

   Where the “provider” in “provider:1500” is the name of your bridge mapping.

   Once this change is made and Neutron services restarted VXLAN networks should show an MTU of 1500.

```
$ os network show test-vxlan | grep mtu
| mtu                         | 1500
```

New instances will get an MTU of 1500. Old instances should get it if their dhcp client is restarted or they are rebooted.

```
# After a reboot
$ ip ad sh ens3
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group defaul
    link/ether fa:16:3e:39:e5:c7 brd ff:ff:ff:ff:ff:ff
    inet 10.50.0.18/24 brd 10.50.0.255 scope global ens3
       valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fe39:e5c7/64 scope link
       valid_lft forever preferred_lft forever
```

Conclusion

Overall, it would just be easier on everyone if the default MTU ended up being 1500 in Neutron VXLAN networks, so I think everyone should make a change like this to their OpenStack cloud. Otherwise inevitably things will fail in weird ways, if for no other reason then everything seems to expect an MTU of 1500.

Find me on twitter **@ccollicutt**

On the web