

虚拟网卡实战

版权说明

本“比特就业课”项目（以下简称“本项目”）的所有内容，包括但不限于文字、图片、音频、视频、软件、程序、数据库、设计、布局、界面等，均由本项目的开发者或授权方拥有版权。我们鼓励个人学习者使用本项目进行学习和研究。在遵守相关法律法规的前提下，个人学习者可以下载、浏览、学习本项目的内容，并为了个人学习、研究或教学目的而使用其中的材料。但请注意，未经我们明确授权，个人学习者不得将本项目的内容用于任何商业目的，包括但不限于销售、转让、许可或以其他方式从中获利。此外，个人学习者也不得擅自修改、复制、传播、展示、表演或制作本项目内容的衍生作品。任何未经授权的使用均属侵权行为，我们将依法追究法律责任。如果您希望以其他方式使用本项目的内容，包括但不限于引用、转载、摘录、改编等，请事先与我们取得联系，获取书面授权。感谢您对“比特就业课”项目的关注与支持，我们将持续努力，为您提供更好的学习体验。特此说明。比特就业课版权所有方。

对比特项目感兴趣，可以联系这个微信。



基础知识

在计算机网络中，tun 与 tap 是操作系统内核中的虚拟网络设备。不同于普通靠硬件网络适配器实现的设备，这些虚拟的网络设备全部用软件实现，并向运行于操作系统上的软件提供与硬件的网络设备完全相同的功能。

tun 是网络层的虚拟网络设备，可以收发第三层数据报文包，如 IP 封包，因此常用于一些点对点 IP 隧道，例如 OpenVPN，IPSec 等。

tap 是链路层的虚拟网络设备，等同于一个以太网设备，它可以收发第二层数据报文包，如以太网数据帧。Tap 最常见的用途就是做为虚拟机的网卡，因为它和普通的物理网卡更加相近，也经常用作普通机器的虚拟网卡。

基础命令

1. 添加网卡

```
Shell
# 创建 tap
ip tuntap add dev tap0 mode tap
# 创建 tun
ip tuntap add dev tun0 mode tun
```

2. 删 除网卡

```
Shell
# 删除 tap
ip tuntap del dev tap0 mode tap
# 删除 tun
ip tuntap del dev tun0 mode tun
```

3. 激活网卡

```
Shell
ip link set tun0 up
```

4. 设置 ip

```
Shell
ip addr add 10.5.0.1/24 dev tun0
```

5. 查看帮助

```
Shell
ip tuntap help
```

实战目的

了解什么是虚拟网卡。

实战步骤

1. 查看当前网卡，其中 eth0 是我们的物理网卡

```
Shell
root@139-159-150-152:~# ifconfig
br-df863876204e: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.0.1 netmask 255.255.0.0 broadcast
192.168.255.255
        inet6 fe80::42:20ff:feb7:55bb prefixlen 64 scopeid
0x20<link>
            ether 02:42:20:b7:55:bb txqueuelen 0 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 29 bytes 4216 (4.2 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast
172.17.255.255
        inet6 fe80::42:d7ff:fe87:d11b prefixlen 64 scopeid
0x20<link>
            ether 02:42:d7:87:d1:1b txqueuelen 0 (Ethernet)
            RX packets 586466 bytes 36737314 (36.7 MB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 860320 bytes 1655179375 (1.6 GB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.183 netmask 255.255.255.0 broadcast
192.168.0.255
        inet6 fe80::f816:3eff:fe9d:f4ac prefixlen 64 scopeid
0x20<link>
            ether fa:16:3e:9d:f4:ac txqueuelen 1000 (Ethernet)
            RX packets 4031203 bytes 4359821919 (4.3 GB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 2642059 bytes 470413180 (470.4 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
        RX packets 1125305 bytes 449342149 (449.3 MB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 1125305 bytes 449342149 (449.3 MB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lxcbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 10.0.3.1 netmask 255.255.255.0 broadcast 10.0.3.255
        inet6 fe80::216:3eff:fe00:0 prefixlen 64 scopeid
0x20<link>
        ether 00:16:3e:00:00:00 txqueuelen 1000 (Ethernet)
        RX packets 595 bytes 72213 (72.2 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 817 bytes 73558 (73.5 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

2. 我们通过 ip link 命令可以完成虚拟网卡的创建，创建网卡

Shell

```
ip tuntap add dev tun0 mod tun
```

3. 查看网卡信息，新添加的虚拟网卡默认是 DOWN 状态.需要用-a 参数显示

Shell

```
root@139-159-150-152:~# ifconfig -a
br-df863876204e: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.0.1 netmask 255.255.0.0 broadcast
192.168.255.255
        inet6 fe80::42:20ff:feb7:55bb prefixlen 64 scopeid
0x20<link>
        ether 02:42:20:b7:55:bb txqueuelen 0 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 29 bytes 4216 (4.2 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast
172.17.255.255
        inet6 fe80::42:d7ff:fe87:d11b prefixlen 64 scopeid
```

```
0x20<link>
    ether 02:42:d7:87:d1:1b  txqueuelen 0  (Ethernet)
    RX packets 586466  bytes 36737314 (36.7 MB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 860320  bytes 1655179375 (1.6 GB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.0.183  netmask 255.255.255.0  broadcast
192.168.0.255
      inet6 fe80::f816:3eff:fe9d:f4ac  prefixlen 64  scopeid
0x20<link>
    ether fa:16:3e:9d:f4:ac  txqueuelen 1000  (Ethernet)
    RX packets 4031391  bytes 4359857658 (4.3 GB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 2642250  bytes 470440756 (470.4 MB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
      inet 127.0.0.1  netmask 255.0.0.0
      inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 1125373  bytes 449348155 (449.3 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1125373  bytes 449348155 (449.3 MB)
        TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lxcbr0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
      inet 10.0.3.1  netmask 255.255.255.0  broadcast 10.0.3.255
      inet6 fe80::216:3eff:fe00:0  prefixlen 64  scopeid
0x20<link>
    ether 00:16:3e:00:00:00  txqueuelen 1000  (Ethernet)
    RX packets 595  bytes 72213 (72.2 KB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 817  bytes 73558 (73.5 KB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

tun0: flags=4240<POINTOPOINT,NOARP,MULTICAST>  mtu 1500
      unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
txqueuelen 500  (UNSPEC)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

4. 激活网卡

```
Shell  
ip link set tun0 up
```

5. 再次 ifconfig 可以看到网卡了

```
Shell  
root@139-159-150-152:~# ifconfig  
tun0: flags=4241<UP,POINTOPOINT,NOARP,MULTICAST> mtu 1500  
          unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  
txqueuelen 500  (UNSPEC)  
          RX packets 0 bytes 0 (0.0 B)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 0 bytes 0 (0.0 B)  
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

6. 分配 ip 地址

```
Shell  
ip addr add 10.5.0.1/24 dev tun0
```

7. 查看 ip 地址, ping 该地址是通的

```
Shell  
  
root@139-159-150-152:~# ifconfig  
tun0: flags=4241<UP,POINTOPOINT,NOARP,MULTICAST> mtu 1500  
          inet 10.5.0.1 netmask 255.255.255.0 destination 10.5.0.1  
          unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  
txqueuelen 500  (UNSPEC)  
          RX packets 0 bytes 0 (0.0 B)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 0 bytes 0 (0.0 B)  
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
root@139-159-150-152:~# ping 10.5.0.1  
PING 10.5.0.1 (10.5.0.1) 56(84) bytes of data.  
64 bytes from 10.5.0.1: icmp_seq=1 ttl=64 time=0.037 ms  
64 bytes from 10.5.0.1: icmp_seq=2 ttl=64 time=0.040 ms
```

```
64 bytes from 10.5.0.1: icmp_seq=3 ttl=64 time=0.042 ms
^C
--- 10.5.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2035ms
rtt min/avg/max/mdev = 0.037/0.039/0.042/0.002 ms
```

8. 此时说明我们的虚拟网卡已经添加好了，通过 del 可以删除网卡

```
Shell
ip tuntap del dev tun0 mod tun
```

比特就業課