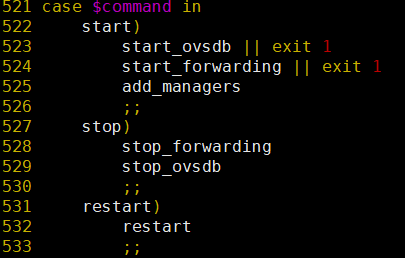
OVS分析

# ovs-vsctl start

sudo /sbin/modprobe openvswitch

sudo /usr/local/share/openvswitch/scripts/ovs-ctl start



steven@ubuntu:~/study/openvswitch-2.12.0$ sudo ./utilities/ovs-ctl start

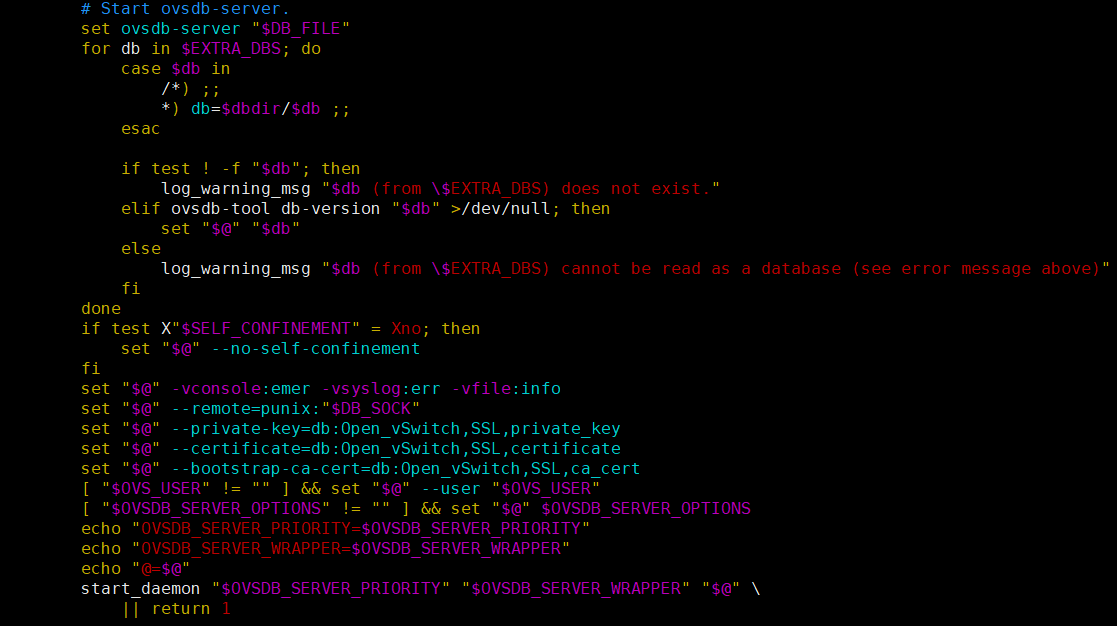
OVSDB\_SERVER=yes

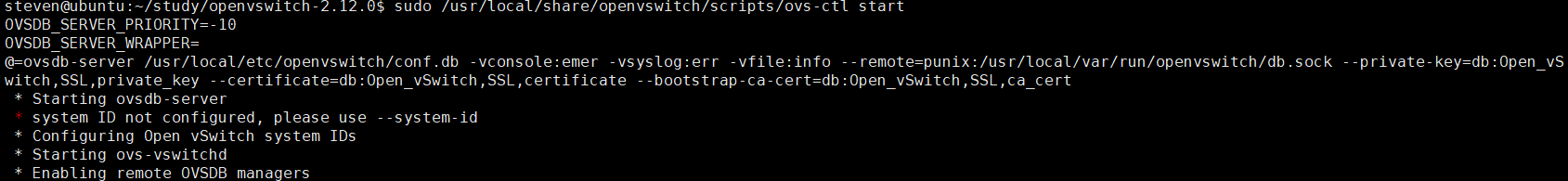
\* Starting ovsdb-server

\* system ID not configured, please use --system-id

\* Configuring Open vSwitch system IDs

./utilities/ovs-ctl: 519: ./utilities/ovs-ctl: ./utilities/ovs-kmod-ctl: not found





等价于执行了下面这句

start\_daemon -10 ovsdb-server /usr/local/etc/openvswitch/conf.db -vconsole:emer -vsyslog:err -vfile:info --remote=punix:/usr/local/var/run/openvswitch/db.sock --private-key=db:Open\_vSwitch,SSL,private\_key --certificate=db:Open\_vSwitch,SSL,certificate --bootstrap-ca-cert=db:Open\_vSwitch,SSL,ca\_cert

RFC 7047 specifies the OVSDB protocol but it does not specify an on-disk storage format.

# ovsdb创建

ovsdb-tool create /usr/local/etc/openvswitch/conf.db

/usr/local/share/openvswitch/vswitch.ovsschema

gdb --args ovsdb-tool create conf.db /usr/local/share/openvswitch/vswitch.ovsschema

# ovs-vswitchd

alias run\_ovsdb-server='sudo ovsdb-server /usr/local/etc/openvswitch/conf.db -vconsole:emer -vsyslog:err -vfile:info --remote=punix:/usr/local/var/run/openvswitch/db.sock --private-key=db:Open\_vSwitch,SSL,private\_key --certificate=db:Open\_vSwitch,SSL,certificate --bootstrap-ca-cert=db:Open\_vSwitch,SSL,ca\_cert --no-chdir --log-file=/usr/local/var/log/openvswitch/ovsdb-server.log --pidfile=/usr/local/var/run/openvswitch/ovsdb-server.pid --detach --monitor'

alias gdb\_ovs-vswitchd='sudo gdb --args /home/steven/study/openvswitch-2.12.0/vswitchd/ovs-vswitchd unix:/usr/local/var/run/openvswitch/db.sock -vconsole:emer -vsyslog:err -vfile:info --mlockall'

# datapath

./configure CFLAGS='-g -O0'

./configure --with-linux=/lib/modules/`uname -r`/build CFLAGS='-g -O0'

./configure --with-linux=/home/steven/work/linux-4.1.30 CFLAGS='-g -O0'

编译错误

/usr/src/linux-headers-4.15.0-99/include/net$ vi dst\_ops.h

sudo /sbin/modprobe openvswitch

/sbin/lsmod | grep openvswitch

sudo /sbin/modprobe -r openvswitch //删除模块；删除模块时需要加‘-r’参数

在datapath/datapath.c中会调用module\_init(dp\_init);来初始化内核模块

Pr\_info的日志在文件/var/log/kern.log

tail -f kern.log 查看日志

参考Linux内核收包流程，函数调用链是：

硬件中断 -->do\_IRQ–>handle\_irq–>e1000\_intr\_msix\_rx–>\_\_napi\_schedule(&adapter->napi)–>

\_\_\_\_napi\_schedule–>\_\_raise\_softirq\_irqoff(NET\_RX\_SOFTIRQ)

do\_IRQ–>irq\_exit–>do\_softirq–>call\_softirq–>\_\_do\_softirq–>

net\_rx\_action->e1000e\_poll–>e1000\_receive\_skb->napi\_gro\_receive–>

netif\_receive\_skb–>\_\_netif\_receive\_skb–>\_\_netif\_receive\_skb\_core–>

deliver\_skb–>ip\_rcv–>NF\_HOOK(NF\_INET\_PRE\_ROUTING)–>

ip\_rcv\_finish–>dst\_input–>ip\_local\_deliver–>

NF\_HOOK(NF\_INET\_LOCAL\_IN)–>ip\_local\_deliver\_finish–>ipprot->handler()

如果ovs 或者普通的网桥如何报文在哪里接收呢？通过梳理代码发现上述调用流程中的\_\_netif\_receive\_skb\_core函数有个rx\_handler，ovs bridge等都是通过设置该函数指针实现对报文的接管。

梳理流程发现ovs的rx\_handler指针是netdev\_frame\_hook；bridge的为br\_handle\_frame

<https://blog.csdn.net/kklvsports/article/details/84136047>

<https://www.cnblogs.com/liuhongru/p/11134435.html>

(gdb) bt

#0 dpif\_netlink\_port\_add\_\_ (dpif=0x9a9520, name=0x9a4f90 "ubuntu\_br", type=OVS\_VPORT\_TYPE\_INTERNAL, options=0x0, port\_nop=0x7fffffffe0b8) at lib/dpif-netlink.c:748

#1 0x00000000005cb71b in dpif\_netlink\_port\_add\_compat (dpif=0x9a9520, netdev=0x9cafd0, port\_nop=0x7fffffffe0b8) at lib/dpif-netlink.c:876

#2 0x00000000005cb8c9 in dpif\_netlink\_port\_add (dpif\_=0x9a9520, netdev=0x9cafd0, port\_nop=0x7fffffffe0b8) at lib/dpif-netlink.c:922

#3 0x00000000004b2db1 in dpif\_port\_add (dpif=0x9a9520, netdev=0x9cafd0, port\_nop=0x7fffffffe120) at lib/dpif.c:584

#4 0x000000000043cca5 in port\_add (ofproto\_=0x9a80c0, netdev=0x9cafd0) at ofproto/ofproto-dpif.c:3721

#5 0x000000000041ffc2 in ofproto\_port\_add (ofproto=0x9a80c0, netdev=0x9cafd0, ofp\_portp=0x7fffffffe260) at ofproto/ofproto.c:2013

#6 0x000000000040b430 in iface\_do\_create (br=0x9a7a70, iface\_cfg=0x9a5d00, ofp\_portp=0x7fffffffe260, netdevp=0x7fffffffe270, errp=0x7fffffffe278)

at vswitchd/bridge.c:1811

#7 0x000000000040b5e3 in iface\_create (br=0x9a7a70, iface\_cfg=0x9a5d00, port\_cfg=0x9a6410) at vswitchd/bridge.c:1849

#8 0x0000000000408ad5 in bridge\_add\_ports\_\_ (br=0x9a7a70, wanted\_ports=0x9a7b50, with\_requested\_port=false) at vswitchd/bridge.c:937

#9 0x0000000000408b67 in bridge\_add\_ports (br=0x9a7a70, wanted\_ports=0x9a7b50) at vswitchd/bridge.c:953

#10 0x0000000000407fa6 in bridge\_reconfigure (ovs\_cfg=0x97b990) at vswitchd/bridge.c:667

#11 0x000000000040f0a7 in bridge\_run () at vswitchd/bridge.c:3044

#12 0x0000000000414b70 in main (argc=6, argv=0x7fffffffe568) at vswitchd/ovs-vswitchd.c:127