ANS User Guide

LIST

➢ ANS Compile

.1. Compile DPDK

➢ Create work directory

```
# mkdir work
```

➢ Download DPDK package to work directory

```
# wget http://dpdk.org/rel/dpdk-17.11.4.tar.xz
ss
```

➢ Uncompressing DPDK package

```
# xz -d dpdk-17.11.4.tar.xz
# tar xvf dpdk-17.11.4.tar
```

➢ Compile all DPDK libs

```
# make config T=x86_64-native-linuxapp-gcc
# make install T=x86_64-native-linuxapp-gcc
```

All DPDK libs are copied to x86_64-native-linuxapp-gcc/lib/ directory.
For detail steps, please refer to DPDK website.
(http://dpdk.org/doc/guides/linux_gsg/index.html）.

Notes: should choice DPDK version based on ANS version.

.2. Generate ANS static libs

➢ Set DPDK environment

```
# export RTE_SDK=/home/work/dpdk-17.11.4
# export RTE_TARGET=x86_64-native-linuxapp-gcc
```

➢ Set ANS environment

```
# export RTE_ANS=/home/work/dpdk-ans
```

➢ Clone ANS from github

```
# git clone https://github.com/ansyun/dpdk-ans.git
```

➢ Generate librte_ans/librte_anssock/librte_anscli

```
# ./ install_deps.sh
librte_ans is generated in librte_ans directory.
```

```
librte_anssock is generated in librte_anssock directory.
librte_anscli is generated in librte_anscli directory.
```

.3. Compile ANS

```
# cd dpdk-ans/ans
# make
```

Notes: If compile ans failed, shall upgrade your gcc and binutils version.

➢ ANS Startup

➢ Run dpdk-setup.sh script to set DPDK environment. Need choice [17], [19], [20], [23].

```
/src/dpdk-17.11.4# ./usertools/dpdk-setup.sh
----------------------------------------------------------------------
---------
 RTE_SDK exported as /root/src/dpdk-17.11.4
----------------------------------------------------------------------
---------
----------------------------------------------------------
 Step 1: Select the DPDK environment to build
----------------------------------------------------------
[1] arm64-armv8a-linuxapp-clang
[2] arm64-armv8a-linuxapp-gcc
[3] arm64-dpaa2-linuxapp-gcc
[4] arm64-dpaa-linuxapp-gcc
[5] arm64-thunderx-linuxapp-gcc
[6] arm64-xgene1-linuxapp-gcc
[7] arm-armv7a-linuxapp-gcc
[8] i686-native-linuxapp-gcc
[9] i686-native-linuxapp-icc
[10] ppc_64-power8-linuxapp-gcc
[11] x86_64-native-bsdapp-clang
[12] x86_64-native-bsdapp-gcc
[13] x86_64-native-linuxapp-clang
[14] x86_64-native-linuxapp-gcc
[15] x86_64-native-linuxapp-icc
[16] x86_x32-native-linuxapp-gcc
```

```
----------------------------------------------------------
 Step 2: Setup linuxapp environment
----------------------------------------------------------
[17] Insert IGB UIO module
[18] Insert VFIO module
[19] Insert KNI module
[20] Setup hugepage mappings for non-NUMA systems
[21] Setup hugepage mappings for NUMA systems
[22] Display current Ethernet/Crypto device settings
[23] Bind Ethernet/Crypto device to IGB UIO module
[24] Bind Ethernet/Crypto device to VFIO module
[25] Setup VFIO permissions

----------------------------------------------------------
 Step 3: Run test application for linuxapp environment
----------------------------------------------------------
[26] Run test application ($RTE_TARGET/app/test)
[27] Run testpmd application in interactive mode
($RTE_TARGET/app/testpmd)

----------------------------------------------------------
 Step 4: Other tools
----------------------------------------------------------
[28] List hugepage info from /proc/meminfo

----------------------------------------------------------
 Step 5: Uninstall and system cleanup
----------------------------------------------------------
[29] Unbind devices from IGB UIO or VFIO driver
[30] Remove IGB UIO module
[31] Remove VFIO module
[32] Remove KNI module
[33] Remove hugepage mappings

[34] Exit Script
```

- ANS startup parameters

```
root@ubuntu:~/dpdk-ans/ans# ./build/ans --help
EAL: Detected 12 lcore(s)

Usage: ./build/ans [options]
```

```
EAL common options:
  -c COREMASK        Hexadecimal bitmask of cores to run on
  -l CORELIST        List of cores to run on
                    The argument format is <c1>[-c2][,c3[-c4],...]
                    where c1, c2, etc are core indexes between 0 and 128
  --lcores COREMAP    Map lcore set to physical cpu set
                    The argument format is
                         '<lcores[@cpus]>[<,lcores[@cpus]>...]'
                    lcores and cpus list are grouped by '(' and ')'
                    Within the group, '-' is used for range separator,
                    ',' is used for single number separator.
                    '( )' can be omitted for single element group,
                    '@' can be omitted if cpus and lcores have the same
value
  --master-lcore ID   Core ID that is used as master
  -n CHANNELS        Number of memory channels
  -m MB             Memory to allocate (see also --socket-mem)
  -r RANKS           Force number of memory ranks (don't detect)
  -b, --pci-blacklist Add a PCI device in black list.
                    Prevent EAL from using this PCI device. The argument
                    format is <domain:bus:devid.func>.
  -w, --pci-whitelist Add a PCI device in white list.
                    Only use the specified PCI devices. The argument format
                    is <[domain:]bus:devid.func>. This option can be
present
                    several times (once per device).
                    [NOTE: PCI whitelist cannot be used with -b option]
  --vdev             Add a virtual device.
                    The argument format is <driver><id>[,key=val,...]
                    (ex: --vdev=net_pcap0,iface=eth2).
  -d LIB.so|DIR      Add a driver or driver directory
                    (can be used multiple times)
  --vmware-tsc-map    Use VMware TSC map instead of native RDTSC
  --proc-type        Type of this process (primary|secondary|auto)
  --syslog           Set syslog facility
  --log-level        Set default log level
  -v                Display version information on startup
  -h, --help         This help

EAL options for DEBUG use only:
  --huge-unlink      Unlink hugepage files after init
  --no-huge          Use malloc instead of hugetlbfs
  --no-pci           Disable PCI
  --no-hpet          Disable HPET
  --no-shconf        No shared config (mmap'd files)

EAL Linux options:
  --socket-mem       Memory to allocate on sockets (comma separated
values)
  --huge-dir         Directory where hugetlbfs is mounted
  --file-prefix      Prefix for hugepage filenames
  --base-virtaddr    Base virtual address
  --create-uio-dev   Create /dev/uioX (usually done by hotplug)
  --vfio-intr        Interrupt mode for VFIO (legacy|msi|msix)
  --xen-dom0         Support running on Xen dom0 without hugetlbfs

  -p PORTMASK: hexadecimal bitmask of ports to configure
  -P : enable promiscuous mode
  --config (port,queue,lcore): rx queues configuration
  --no-numa: optional, disable numa awareness
  --enable-kni: optional, disable kni awareness
  --enable-ipsync: optional, sync ip/route from kernel kni interface
  --enable-jumbo: enable jumbo frame which max packet len is PKTLEN in
decimal (64-9600)
```

- ANS startup example

```
# ./build/ans -c 0x4 -n 1 --base-virtaddr=0x2aaa2aa0000 -- -p 0x1
--config="(0,0,2)"
EAL: Detected 12 lcore(s)
EAL: 128 hugepages of size 2097152 reserved, but no mounted hugetlbfs found
for that size
EAL: Probing VFIO support...
EAL: PCI device 0000:06:00.0 on NUMA socket -1
EAL:   probe driver: 8086:10fb net_ixgbe
EAL: PCI device 0000:06:00.1 on NUMA socket -1
EAL:   probe driver: 8086:10fb net_ixgbe
EAL: PCI device 0000:07:00.0 on NUMA socket -1
EAL:   probe driver: 8086:10fb net_ixgbe
EAL: PCI device 0000:07:00.1 on NUMA socket -1
EAL:   probe driver: 8086:10fb net_ixgbe
param nb 1 ports 1
port id 0
```

- ANS startup with kni/ipsync enable

```
# ./build/ans -c 0x4 -n 1 --base-virtaddr=0x2aaa2aa0000 -- -p 0x1
--config="(0,0,2)" --enable-kni --enable-ipsync
```

➢ ANS Configuration

➢ Compile anscli

```
# make
```

➢ Run anscli with command directly

```
# ./build/anscli "help"
```

➢ Run anscli

```
# ./build/anscli
ans>
```

➢ Run anscli with file-prefix

```
# ./build/anscli --file-prefix=host
ans>
```

➢ Run anscli with file-prefix and command

```
# ./build/anscli --file-prefix=host "help"
```

Notes：

➢ should run ans process before run anscli.

➢ File-prefix shall same as ans’s file-prefix.

.1. help

```
ans> help
ip addr add IFADDR dev STRING
ip addr del IFADDR dev STRING
ip addr show
ip route add DESTIP via NEXTHOP
ip route del DESTIP
ip route show
ip link show
ip neigh show
ip stats show
acl add index NUMBER srcaddr IPADDR dstaddr IPADDR srcportstart NUMBER
srcportend NUMBER dstportstart NUMBER dstportend NUMBER protocol NUMBER
dev IFACE
   index - ACL rule index [1 - 2048], large index has high priority.
   srcaddr  -  source  IP  subnet  address,  0.0.0.0/0  match  all  IP,
[ip-address/mask]
```

```
  dstaddr - destination IP subnet address, 0.0.0.0/0 match all IP,
[ip-address/mask]
  srcportstart - source port start [0...65535]
  srcportend - source port end [0...65535]
  dstportstart - destination port start [0...65535]
  dstportend - destination port start [0...65535]
  protocol - IP protocol, 0 match all protocol, [0...255]
  iface - input interface name, 'any' match all iface
  drop|accept - drops or accepts all packets that match the rule
 note: match ACL rule at PREROUTING.
acl del index NUMBER
  index - ACL rule index [1 - 2048]
acl show
bypass add...
bypass del...
  protocol - IP protocol, 0 match all protocol, [0...255]
  srcport - source port, 0 match all source port, [0...65535]
  dstport - destination port, 0 match all destination port, [0...65535]
 note: match bypass rule at PREROUTING.
       bypass: forward packets to kernel.
bypass show
flow filter add ...
flow filter del ...
  portid - DPDK port id
  dstip - destination IP address, 0.0.0.0: disable destination IP filter
  dstport - destination port, 0: disable destination port filter
  queueid - RX queue id of the DPDK port
 note: match the rule traffic will be forwarded the queue.
flow filter show
port queue show
log level set [emerg | alert | crit | err | warning | notice | info | debug]
help
quit
ans>
```

.2. Configure IP

➢ Add IP

```
ans> ip addr add 10.10.10.10/24 dev veth0
Add IP address successfully
ans>
```

➢ Delete IP

```
ans> ip addr del 10.10.10.10/24 dev veth0
Del IP address successfully
ans>
```

- Show IP

```
ans> ip addr show

 eth0: mtu 1500
   link/ether 08:00:27:de:5d:8e
   inet addr: 10.0.0.2/24
ans>
```

.3. Configure route

- Add route

```
ans> ip route add 20.0.0.0/24 via 10.0.0.20
Add routing successfully
ans>
```

- Delete route

```
ans> ip route del 20.0.0.0/24
Del routing successfully
ans>
```

- Show route

```
ans> ip route show

ANS IP routing table
 10.0.0.0/24 via dev veth0 src 10.0.0.2
 10.10.0.0/24 via 10.0.0.5 dev veth0
ans>
```

.4. Configure neigh

- Show arp table

```
ans> ip neigh show

ANS IP neigh table
   10.0.0.11 dev veth0 lladdr 08:00:27:82:ca:ad REACHABLE
ans>
```

.5. Configure link

- Show link status

```
ans> ip link show

  veth0: port 0 state UP speed 1000Mbps full-duplex mtu 1500
      link/ether 08:00:27:de:5d:8e
      RX packets:29 errors:0 dropped:0
      TX packets:4 errors:0 dropped:0
      RX bytes:5433 TX bytes:312
ans>
```

.6. Show IP statistics

```
ans> ip stats show
  Total packets received                    :33
  Checksum bad                              :0
  Packet too short                          :0
  Not enough data                           :0
  IP header length < data size              :0
  IP length < ip header length              :0
  Fragments received                        :0
  Frags dropped (dups, out of space)        :0
  Fragments timed out                       :0
  Packets forwarded                         :0
  Packets fast forwarded                    :0
  Packets rcvd for unreachable dest         :0
  Packets forwarded on same net             :0
  Unknown or unsupported protocol           :0
  Datagrams delivered to upper level        :31
  Total ip packets generated here           :3
  Lost packets due to nobufs, etc.          :0
  Total packets reassembled ok              :0
  Datagrams successfully fragmented         :0
  Output fragments created                  :0
  Don't fragment flag was set, etc.         :0
  Error in option processing                :0
  Packets discarded due to no route         :0
  IP version != 4                           :0
  Total raw ip packets generated            :0
  IP length > max ip packet size            :0
  Multicasts for unregistered grps          :0
  No match gif found                        :0
  Invalid address on header                    :0
  Packets filtered                              :0
ans>
```

.7. Configure ACL

➢ Add acl rule

```
ans> acl add index 100 srcaddr 10.10.10.0/24 dstaddr 20.20.20.0/24 srcportstart 0 srcportend 65535 dstportstart 0 dstportend 65535 protocol 0 iface any drop
Add ACL rule successfully
ans>
```

➢ Delete acl rule

```
ans> acl del index 100
Delete ACL rule successfully
ans>
```

➢ Show acl rule

```
ans> acl show

ACL rule 100:
    Source subnet address          : 10.10.10.0/24
    Destination subnet address : 20.20.20.0/24
    Source port range                 : 0 - 65535
    Destination port range         : 0 - 65535
    IP protocol                          : 0
    Interface name                     : any
    Action                                  : drop
ans>
```

.8. Configure bypass

➢ Add bypass rule

```
ans> bypass add protocol 17 dstport 68
Add bypass rule successfully
ans>
```

➢ Delete bypass rule

```
ans> bypass del protocol 17 dstport 68
Del bypass rule successfully
```

```
ans>
```

- Show bypass rule

```
ans> bypass show

Bypass rule 0:
    IP protocol                : 17
    Destination port       : 68
ans>
```

.9. Show port queue

- Show port queue lcore mapping

```
ans> port queue show
                port          queue       lcore
                0               0             1
                0               1             2
ans>
```

.10. Configure flow

- Add flow filter rule

```
ans> flow filter add portid 0 dstip 10.0.0.2 dstport 80 queueid 1
Add flow filter successfully
ans>
```

- Delete flow filter rule

```
ans> flow filter del portid 0 dstip 10.0.0.3 dstport 80 queueid 0
Del flow filter rule successfully
ans>
```

- Show flow filter rule

```
ans> flow filter show

Flow filter rule 0:
    Port ID                          : 0
    Destination IP               : 10.0.0.2
    Destination port            : 80
    Queue ID                         : 1
```

```
Flow filter rule 1:
    Port ID                 : 0
    Destination IP          : 10.0.0.3
    Destination port        : 80
    Queue ID                : 0
ans>
```

➢ ANS IP synchronization

.1. IP synchronization enables

```
# ./build/ans  -c 0x2 -n 1 --base-virtaddr=0x2aaa2aa0000 -- -p 0x1 --config="(0,0,1)" --enable-kni
--enable-ipsync
EAL: Detected 2 lcore(s)
EAL: Probing VFIO support...
EAL: PCI device 0000:00:03.0 on NUMA socket -1
EAL:    probe driver: 8086:100e net_e1000_em
```

ANS shall create veth0 interface in linux kernel and ans stack.

```
# ip addr show

6: veth0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:de:5d:8e brd ff:ff:ff:ff:ff:ff
```

.2. KNI interface up

```
# ifconfig veth0 up
# ifconfig
veth0     Link encap:Ethernet  HWaddr 08:00:27:de:5d:8e
          inet6 addr: fe80::a00:27ff:fede:5d8e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:648 (648.0 B)
```

.3. Configure bypass rule

```
ans> bypass add protocol 17 dstport 68
Add bypass rule successfully
ans>
```

.4. Configure IP by dhcp or manual

- Linux side

```
# dhclient -i veth0
# ip addr add 10.10.0.20/24 dev veth0
6: veth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN
group default qlen 1000
    link/ether 08:00:27:de:5d:8e brd ff:ff:ff:ff:ff:ff
    inet 192.168.10.117/24 brd 192.168.10.255 scope global veth0
       valid_lft forever preferred_lft forever
    inet 10.10.0.20/24 scope global veth0
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fede:5d8e/64 scope link
       valid_lft forever preferred_lft forever

i# ip route show
default via 192.168.10.1 dev veth0
10.0.0.0/24 dev enp0s8  proto kernel  scope link  src 10.0.0.11
10.10.0.0/24 dev veth0  proto kernel  scope link  src 10.10.0.20
192.168.10.0/24 dev enp0s3  proto kernel  scope link  src 192.168.10.109
192.168.10.0/24 dev veth0  proto kernel  scope link  src 192.168.10.117
192.168.56.0/24 dev enp0s10  proto kernel  scope link  src 192.168.56.20
```

- ANS side

```
# ./build/anscli
ans> ip addr show

 veth0: mtu 1500
   link/ether 08:00:27:de:5d:8e
   inet addr: 10.0.0.2/24
   inet addr: 192.168.10.117/24
   inet addr: 10.10.0.20/24

ans> ip route show

ANS IP routing table
```

```
  0.0.0.0/0 via 192.168.10.1 dev veth0
  10.0.0.0/24 via dev veth0 src 10.0.0.2
  10.10.0.0/24 via 10.0.0.5 dev veth0
  192.168.10.0/24 via dev veth0 src 192.168.10.117
ans>
```