



Netronome TrafGen

NETRONOME TRAFGEN IS
A STANDALONE UTILITY
FOR SOURCING AND
SINKING NETWORK
TRAFFIC.

CONTENTS

INTRODUCTION.....	1
INSTALLATION.....	1
OPTIONS.....	2
USAGE MODES.....	3
BENCHMARKING TRAFFIC PATTERN	4
REFERENCES	5

INTRODUCTION

Netronome TrafGen is a standalone utility for sourcing and sinking network traffic. TrafGen is a DPDK-based application that is distributed as part of the Agilio software package.

INSTALLATION

TrafGen is shipped as part of the Agilio software distribution and fully integrated in the build environment. It must be compiled on the host system before it can be used. The makefile that is provided assumes a standard Agilio software build environment is present so there is no need to set the RTE_SDK environment variable.

To build the TrafGen applications, simply change to the appropriate directory and execute the makefile as shown in Figure 1 below.

```
root@ubuntu:~# cd /opt/netronome/samples/dpdk/TrafGen
root@ubuntu:~# make
  CC main.o
  LD TrafGen
  INSTALL-APP TrafGen
  INSTALL-MAP TrafGen.map
root@ubuntu:~#
```

Figure 1. Build TrafGen utility



Installing TrafGen in a Virtual Machine (VM)

After the initial Agilio software software has been installed, the TrafGen utility can also be installed in a VM by copying the source directory into the VM and following the same procedure described above.

OPTIONS

Executing the TrafGen utility without any command line arguments will give a complete list of available options. The list of options is broken into two groups. The first is a list of EAL options that are common to all DPDK based applications. The second group of options are application specific.

Note that when executing the application and specifying command line options, the EAL arguments must be specified first, followed by a double hyphen “--” and then the application specific options. Both groups of options are described below.

DPDK EAL options

The EAL options are thoroughly described in the standard DPDK documentation. Two options require Agilio software specific values. They are `-d` which specifies the Agilio software specific poll mode driver and `-w/--whitelist` which specify the PCI address of the VFs to be used.

EAL common options:

```

-c COREMASK : A 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
--master-lcore ID: Core ID that is used as master
-n NUM      : Number of memory channels
-v          : Display version information on startup
-m MB      : memory to allocate (see also --socket-mem)
-r NUM     : force number of memory ranks (don't detect)
--syslog   : set syslog facility
--log-level : set default log level
--proc-type : type of this process
--pci-blacklist, -b: add a PCI device in black list.
               Prevent EAL from using this PCI device. The argument
               format is <domain:bus:devid.func>.
--pci-whitelist, -w: 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=eth_pcap0,iface=eth2).
--vmware-tsc-map: use VMware TSC map instead of native RDTSC

```

EAL options for DEBUG use only:

```

--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:

```

-d LIB.so : add driver (can be used multiple times)

```



```

--xen-dom0 : support application running on Xen Domain0 without
hugetlbfs
--socket-mem : memory to allocate on specific
                sockets (use comma separated values)
--huge-dir   : directory where hugetlbfs is mounted
--file-prefix: prefix for hugepage filenames
--base-virtaddr: specify base virtual address
--vfio-intr: specify desired interrupt mode for VFIO (legacy|msi|msix)
--create-uio-dev: create /dev/uioX (usually done by hotplug)
    
```

Figure 2. DPDK EAL options

Application Specific Options

==== Application Usage =====

```

./build/TrafGen [EAL options] -- [APP options]
Mandatory APP arguments:
--portmask|-p PORTMASK: hexadecimal bitmask of ports to use
APP mode select. Default is drop, or select one of the following:
--benchmark|-b: Benchmark mode (transmit, drop and summarize)
--forward|-f: Forward packets unmodified
--echo|-e: Echo packets with src and dst mac swapped
--log|-l: Log and drop packets to TrafGen.pX.log (where X is the port)
--asymmetric|-a: Forward packets to the highest-numbered port
Optional APP arguments:
--queues-per-lcore|-q NQ: number of queue (=ports) per lcore (default
1)
--stats-period|-T PERIOD: statistics will be refreshed each PERIOD
seconds (0 to disable, default 1, 86400 maximum)
--runtime|-Q COUNT: quit after COUNT periods, 0 to disable (default)
--summary|-s SUMMARY.TXT: save the summary statistics to a file
--src-ip|-u SRC_IP: base source IP address
--dst-ip|-v DST_IP: base destination IP address
--ip-stride|-k IP_STRIDE: IP stride between streams (default 1.0.0.0)
--src-mac|-w SRC_MAC: base source MAC address
--dst-mac|-x DST_MAC: base destination MAC address
--mac-stride|-g MAC_STRIDE: MAC stride between streams (default
01:00:00:00:00:00)
--streams|-i NUM_STREAMS: number of streams (default 1)
--flows-per-stream|-y NUM_FLOWS: flows per stream (default 65536)
--bursts-per-stream|-j NUM_BURSTS: number of bursts per stream (de-
fault 1)
--packet-size|-z PKT_SIZE: packet size (0 for IMIX, default 64)
--tx-burst|-t TX_BURST_SIZE: TX burst size (default 32)
--pps|-r TX_PPS: TX packet rate to attempt (default 0)
    
```

Figure 3. Application Options

USAGE MODES

TrafGen has 6 main modes of usage. Only one mode may be specified at a time.

- Drop [default - no mode option specified]: Count and drop all received traffic
- Benchmark [**--benchmark|-b**]: Benchmark mode (transmit, drop and summarize)
- Forward [**--forward|-f**]: Forward packets unmodified
- Echo [**--echo|-e**]: Echo packets with src and dst mac swapped
- Log [**--log|-l**]: Log and drop packets to TrafGen.pX.log (where X is the port)
- Asymmetric [**--asymmetric|-a**]: Forward packets to the highest-numbered port



```

=====
| Timer period:                                     1 |
+----- Statistics for port 0 -----+
| Packets sent:                                   49018784 |
| Packets received:                              10780203 |
| Packet receive rate:                          1888517 |
| Packet send rate:                              9012128 |
| Bytes received:                               689933056 |
| Byte receive rate:                            120865088 |
| Packets dropped on send:                       0 |
| Packets dropped on receive:                   10780207 |
+----- Packet Generation -----+
| Short transmit bursts:                        0 |
| Generate overrun:                            0 |
| Allocation failures:                          0 |
=====
Timestamp: 1452885156
    
```

Figure 4. Running TrafGen benchmark mofe with 2 VDs

BENCHMARKING TRAFFIC PATTERN

TrafGen was created in order to more easily generate traffic patterns for performance benchmark testing. The key parameters for this testing are:

- --flows-per-stream
- --bursts-per-stream
- --streams
- --packet size

A main benchmark is to generate 64000 microflows of 64B with 4 VMs. The traffic is broken down into 32 streams and 10 bursts per stream. Since there are 4 VMs, the total number streams per VM is 8 (32 streams/4 VMs) and the total number of flows per stream is 2000 (64000 microflows/32 streams).

Each VM instance of TrafGen is represented in Figure 5 below.

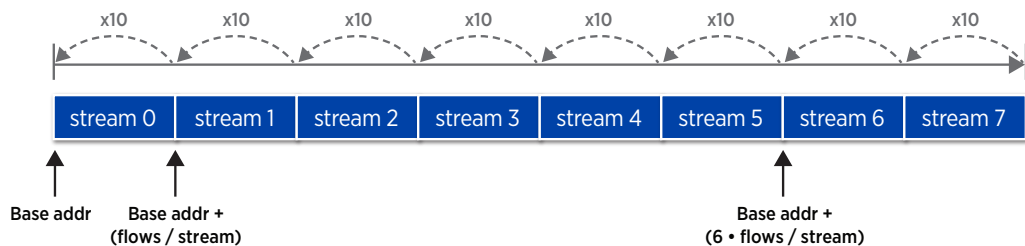


Figure 5. Benchmarking traffic profile

An example command line for the diagram above is shown in Figure 6:

```

root@ubuntu:~# /opt/netronome/samples/dpdk/TrafGen/build/TrafGen -c 0x3
-n 4 --socket-mem 256 \
-d /opt/netronome/lib/librte_pmd_nfp_net.so -w 0000:00:08.0
-- --benchmark --portmask 0x1 --packet-size 64 --streams 8 --flowper-
stream 2000 --burstsperstream 10
    
```

Figure 6. Build TrafGen utility



REFERENCES

- /opt/netronome/samples/dpdk/TrafGen/README
- /opt/netronome/samples/dpdk/TrafGen/sample-usage.sh
- DPDK.org
- Agilio software documentation