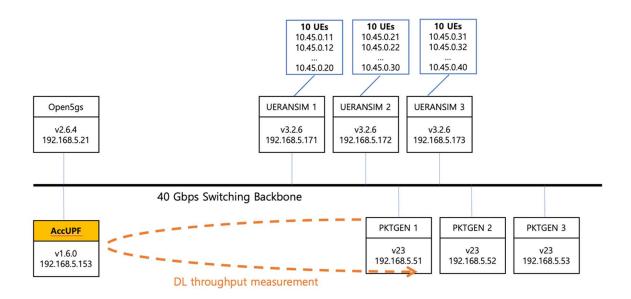
Title: A Demonstration of 30-Gbps Load Testing for Accelerated UPF with Open5gs

Introduction

AccUPF is an accelerated UPF supported by DPDK, NextEPC Inc. has released v1.6 on September 2023. In this document, we demonstrate how to test its performance using Open5gs, Pktgen and UERANSIM.

Network Configuration

The 5G core is composed of Open5gs Open Source v2.6.4 and AccUPF v1.6. We install UERANSIM v3.2.6 which emulates three gNodeB with 30 active UEs. Since the UERANSIM is not suitable to send and receive a large amount of data packets, we separately install Pktgen v23 to generate a large amount of packets accelerated by DPDK.



Hardware Specification for AccUPF

Server Model	Dell PowerEdge R450	
CPU	2x Intel Xeon Gold 5315Y 3.2GHz 8 cores	
Memory	32 GB	
Storage	480 GB SSD	
NIC	Supermicro AOC-S40G-i2Q	
	- 2 QSFP+ ports	
	- 40 Gbps per port	
	- PCI Express 3.0	

	-	Intel XL710 40GbE controller

Test Plan

Our goal is to demonstrate the performance of AccUPF. Since each Pktgen virtual machine has a dedicated 10Gbps NIC, we could generate around 30Gbps UDP traffic at max in the 5G downlink toward UEs. We test downlink only, because the uplink requires to generate GTP traffic which is not supported by Pktgen.

Test Procedure

- 1. Run Open5gs and AccUPF
 - a. Add 30 subscribers in Open5gs
 - b. Please make sure that UE AMBR is equal or greater than 1Gbps
- 2. Run UERANSIM to register 30 UEs.
 - a. At this point, AccUPF has all the PDU session information via N4 interface
- 3. Pktgen creates UDP traffic for all 30 UEs
 - a. We use three Pktgen virtual machines and sequence command is utilized to generate multiple traffic flows.
 - b. Each Pktgen generates 10 sequences toward 10 UEs at a rate of 10 Gbps.
 - c. For a single sequence,
 - i. Destination IP: UE IP
 - ii. Destination MAC: MAC address of AccUPF's N6 interface
 - d. For more information on Pktgen, Please refer to the link: <u>https://pktgen-</u><u>dpdk.readthedocs.io/en/latest/getting_started.html</u>
- 4. AccUPF creates GTP encapsulated packets from the incoming data received by N6, and forward them to UERANSIM's gNodeB through N3.
- 5. We want to redirect the GTP packets to Pktgen because
 - a. The amount of traffic is too much to be handled by UERANSIM
 - b. We want to compare Tx/Rx results in Pktgen
- 6. AccUPF has a function to accept static ARP entries. We add static ARP entries for the emulated gNodeBs toward Pktgen.
- 7. All GTP encapsulated packets are delivered to Pktgen.

Test Results

Distance	
Pktgen	
S. 192.168.5.52 (pktgen)	□ ×
👔 Re-attach 🛟 Fullscreen 🚇 Stay on top 👔 Duplicate 🧭 🔍 🍳 🗮 👼 🦸 Hide toolbar	🔀 Close 👔 Re-attach 👯 Fullscreen 📵 Stay on top 👔 Duplicate 🧭 🔍 🥥 📑 👼 🖇 Hide toolbar
/ Ports 0-1 of 2 <main page=""> Copyright(c) <2010-2023>, Intel Corporation</main>	\ Ports 0-1 of 2 <main page=""> Copyright(c) <2010-2023>, Intel Corporation</main>
Port:Flags : 0: Seq 1: Seq Link State : <up-10000-fd> <up-10000-fd>Tot</up-10000-fd></up-10000-fd>	Port:Flags : 0: Seg ! Seg Total Rate
Pkts/s Rx : 0 772,909 772,909	Link State : <up-10000-fd> <up-10000-fd> Total Rate</up-10000-fd></up-10000-fd>
836,785 : 772,767	Pkts/s Rx : 0 816,737 816,737 Tx : 816,909 0 816,808
836,637 0.8.778 TX RX 9.052 0 9.052/8,778 .800/9,504 1.045,666	MBits/s Rx/Tx : 0 9,278 TX RX 9,565 0 9,565/9,278
1,045,632 0 1,045,632	Pkts/s Rx Max : 044,257 844,257
	Tx Max : 844,389 0 844,389 Broadcast : 0 0
Multicast : 0 270 Sizes 64 : 2 272	Broadcast : U U U Multicast : 0 37
512e5 64 : 2 2/2 65-127 : 0 0	Sizes 64 : 1 38
128-255 : 0 0	65-127 : 0 0
256-511 : 0 0	128-255 : 0 0 256-511 : 0 0
512-1023 : 0 0 1024-1518 : 0 221.111.030	512-1023 : 0 0
1024-1518 : 0 221,111,030 Runts/Jumbos : 0/0 0/0	1024-1518 : 0 23.939.390
ARP/ICMP Pkts : 0/0 0/0	Runts/Jumbos : 0/0 0/092.168.4.211/24 5000
	ARP/ICMP Pkts : 0/0 0/092.168.4.211/24 5000
Total Rx Pkts : 2 220,572,748 Tx Pkts : 220.930,176 0	Errors Rx/Tx : 0/0 0/092.168.4.211/24 5000 Total Rx Pkts : 1 23.820.79892.168.4.211/24 5000
TX PKTS : .220,930,170 U Rx/TX MBS : 0/2,509,766 2,583,345/0	Tx Pkts : 24,097,256 092.168.4.211/24 5000
TCP Flags : A	Rx/Tx MBs : 0/273,744 278,988/092.168.4.211/24 5000
TCP Seq/Ack : 74616/74640 74616/74640	TCP FlagsAA92.168.4.211/24 5000 TCP Seg/Ack 74616/74640 74616/7464092.168.4.211/24 5000
Pattern Type : abcd abcd Tx Count/% Rate : Forever /60% Forever /100%	Pattern Type : abcd abcd 21/24 5000
TX COUNT/S Hate : Polevel /005 Polevel /1005 Pkt Size/Rx:TX Burst: 64 / 64: 64	Tx Count/% Rate : Forever /60% Forever /100%92.168.4.211/24 5000
TTL/Port Src/Dest : 64/ 1234/ 5678 64/ 1234/ 5678	Pkt Size/Rx:Tx Burst: 64 / 64: 64 64 / 64: 64
Pkt Type:VLAN ID : IPv4 / UDP:0001 IPv4 / UDP:0001	TTL/Port Src/Dest : 64/ 1234/ 5678 64/ 1234/ 5678 Pktgen 23.06.1 (D:-1/15ad:7b0/0000:02:03-1/15ad:7b0/0000:02:04.0
\02.1p CoS/DSCP/IPP : 0/ 0/ 0 0/ 0/ 0 VxLAN Flg/Grp/vid : 0: Seq 1: Seq	- Friger 25.00.1 (01/1340.700/0000.02.03-1/1340.700/0000.02.04.000000.02.04
TP Destination : <up-10000-fd> <up-10000-fd></up-10000-fd></up-10000-fd>	6. 192.168.5.53 (pktgen)
MAC Destination : 898,669 0 898,669 Source : 0/10,208 10,226/0 10,226/10,208	👔 Re-attach 💱 Fullscreen 🌉 Stay on top 👔 Duplicate 🧭 🔍 🍭 🗮 👼 🗲 Hide toolbar
Source : 0/10,208 10,226/0 10,226/10,208 NUMA/Vend:ID/PCI : 1 1,045,666 1,045,666	102.1p CoS/DSCP/IPP : 0/ 0/ 0 0/ 0/ 0
- Pktgen 23.06.1 (DP 1,045,632 0 1,045,632	VxLAN Fla/Grop/vid : 0: Seg 1: Seg
s set all seq 0 0	IP Destination : <up-10000-fd> <up-10000-fd></up-10000-fd></up-10000-fd>
Pktgen:∕> set 0 src i 0 224 src IP address should 2 226	Source : 0 831,593 831,593
Sic LP address should 2 220 Pktgen: // set 1 src 1 0 0	MAC Destination : 831,616 Source : 019,447 TX RX 9,739 0 9,739/9,447
src IP address should 0 0	NUMA/Vend:ID/PCI : 954.551
Pktgen: /> set all dst 0 0	- Pktgen 23.06.1 (DP 954,549 0 954,549
Pktgen:/> set all dst 0 0 Pktgen:/> set all pro 0 181,283,985	s set all seq 0 0 Pktgen:∕> set 0 src i 0 260
Ntgen:// sequence 0 0/0 0/092.168.4.212/24 5000 6000 ipv4 udp 0 1400	src IP address should 2 262
Pktgen: /> sequence 1 0/0 0/092.168.4.212/24 5000 6000 ipv4 udp 0 1400	Pktgen:/> set 1 src i 0 0
Pktgen: > sequence 2 0/0 0/092.158.4.212/24 5000 6000 ipv4 udp 0 1400	src IP address should 0 0
Pktgen: /> sequence 3 2 181,084,53692.168.4.212/24 5000 6000 ipv4 udp 0 1400 Pktgen: /> sequence 4 181,439,865 092.168.4.212/24 5000 6000 ipv4 udp 0 1400	Pktgen:∥> set all dst 0 0 Pktgen:∥> set all dst 0 0
Pktgen: /> sequence 5 0/2,061,156 2,120,859/092.168.4.212/24 5000 6000 ipv4 udp 0 1400	Pktgen:/>set all pro 0 209,238,123
Pktgen: /> sequence 6 0 3cec:ef3e:15b1 000c:2907:52fa 10.45.0.27 192.168.4.212/24 5000 6000 ipv4 udp 0 1400	Pktgen: /> sequence 0 0/0 0/092.168.4.213/24 5000

The screen shots above are captured by three Pktgen terminals during a 30 Gbps Throughput Test run. The Rx rate is always slightly higher due to the added GTP header.

AccUPF's stat monitor

```
Statistics for port 0 -----
Packets sent:
                            290603620
Packets received:
                                    465
Packets TX pps:
Packets TX Mbps:
Packets TX frag:
Packets RX pps:
Packets RX Mbps:
Packets RX mbps:
                              2531202
                               29159 TX
                                  0
                                    1
                                    0
Packets RX reas:
                                    0
Packets dropped:
                                   461
Packets drop pps:
                                    1
DEV Stats
    DEV RX packets 465 bytes 27900
    DEV RX errors 0 missed 0 no-mbuf 0
    DEV TX packets 290603816 bytes 418469490708
    DEV TX errors 0
Packets drop detail:
    16:[461]
Statistics for port 1 -----
Packets sent: 0
Packets received: 288072564
Packets TX pps: 0
0
Packets TX Mbps:
Packets TX frag:
Packets TX frag:
Packets RX pps:
Packets RX Mbps:
Packets RX reas:
Packets dropped:
                                     0
                                    0
                               2552816
                                 28509 RX
                                    58
Packets drop pps:
                                    0
DEV Stats
    DEV RX packets 288073491 bytes 402150515948
    DEV RX errors 0 missed 0 no-mbuf 0
    DEV TX packets 0 bytes 0
    DEV TX errors 0
Packets drop detail:
    16:[58]
Packets RX / Lcore:
    core10:[76222040] core12:[67749580] core14:[79667216] core16:[64433728]
Aggregate statistics ===================================
Total packets sent: 288072418
Total packets received: 288073028
Total packets dropped: 518
```

The screen shots above are captured by AccUPF's console during a 30 Gbps Throughput Test run. In the figure, Port 0 is the N3 interface while Port 1 is N6 interface.

Conclusion

The test method we showed in this document is scalable as the number of Pktgen instances increases and it is not only for AccUPF, but applicable to any other UPF product. For any question regarding the AccUPF, please contact NextEPC.

Technical Contact: <u>help@nextepc.com</u> Sales Contact: <u>sales@nextepc.com</u>