FRANKFURT
UNIVERSITY
OF APPLIED SCIENCES

Mobile Computing
M.Eng. Information Technology

Winter Semester 24/25

Slicing-UPF-AMF-Traffic

Supervised by
Prof. Dr. Armin Lehmann

FRANKFURT UNIVERSITY OF APPLIED SCIENCES

Team: STAMHS_5G

Name	Matriculation Number
Md. Ashraf Uddin	1398481
Md Mosharraf Hossain	1386448
Md Sohel Rana	1428128
Saleque Ahmed	1393172
Hasibuzzaman	1448140
Tanvir Ahmed	1386435

Table of contents

- ✓ Introduction
- ✓ Open5GS Network Function
- ✓ Network Slicing
- ✓ PacketRusher
- ✓ RAN, GNB & UEs
- ✓ System Architecture
- ✓ Containerized Deployment Overview
- ✓ UE Registration & Session Establishment Testing
- ✓ IP configurations of UPFs and Packetrusher
- ✓ Listening to receive traffic
- ✓ Traffic Throughput for Each UE
- ✓ Traffic Received at UPF
- ✓ UPF Internet Connectivity Test
- ✓ Conclusion
- ✓ References

Introduction

- ✓ Standalone 5G Core Network
- ✓ Simulated 5G RAN & UE Traffic
- ✓ Real Time Monitoring & Traffic Analysis
- ✓ Docker Based Deployment

Open5GS Network Function

- NSSF
- NSR
- UDM
- UDR
- AUSF
- AMF
- PCF
- SMF
- UPF

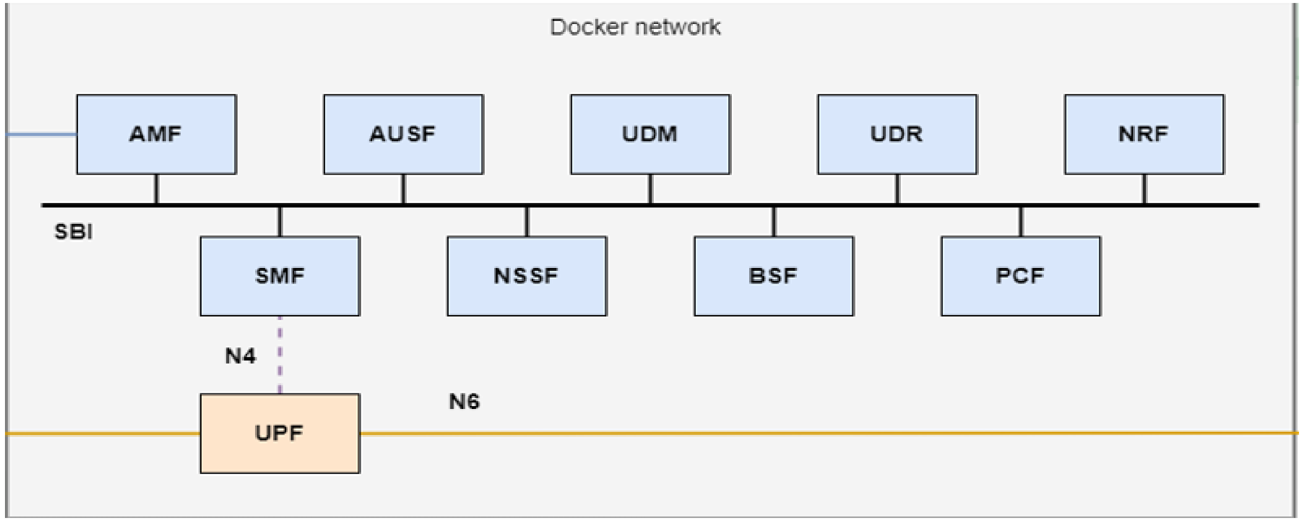


Fig 1: Network Function

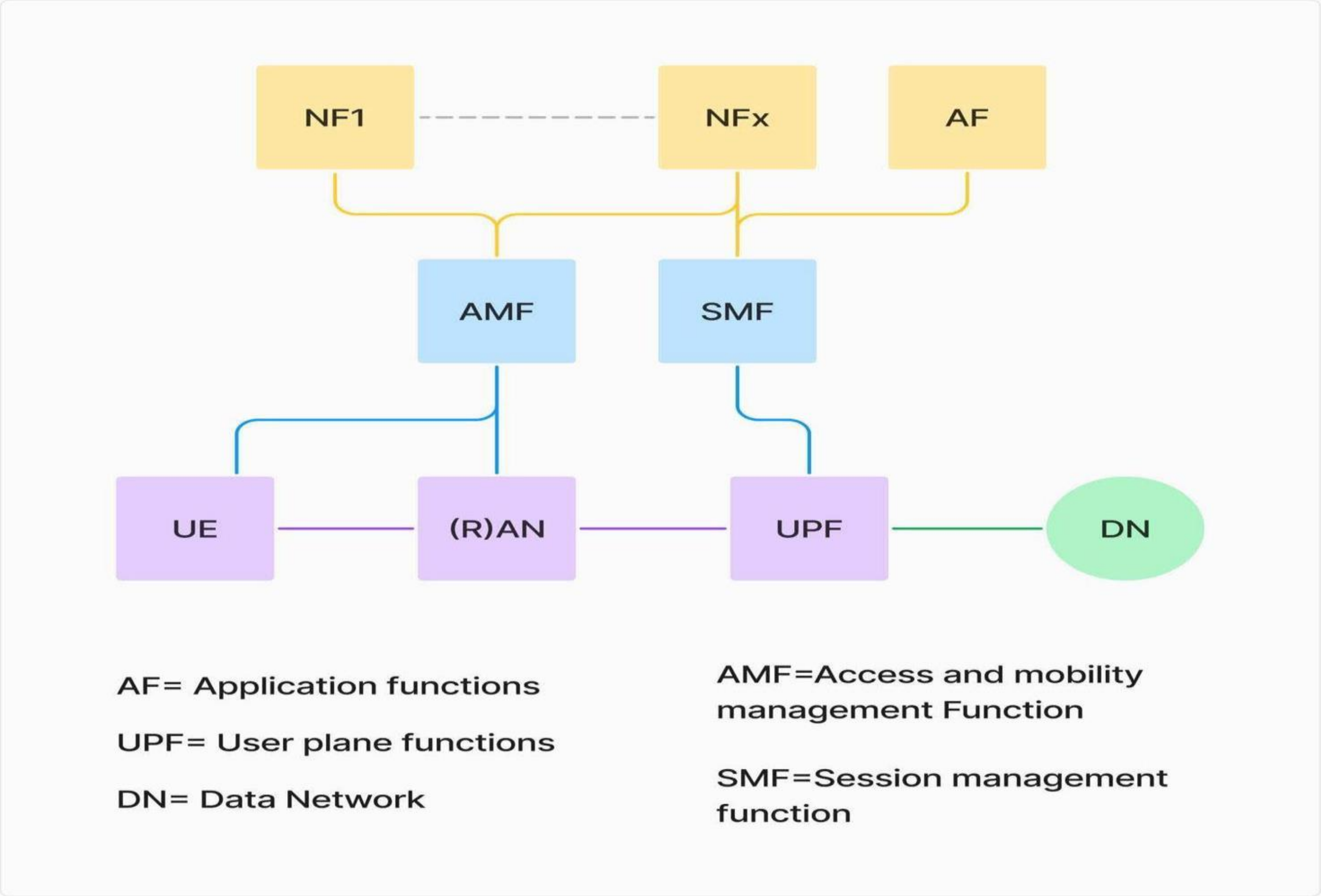
Image Source

Network Slicing

- ✓ Network slicing is a new network architecture that provides multiple logical networks on the same shared network infrastructure.
- ✓ Each slice of the network can be allocated based on the specific needs of the application. This is an essential element of the 5G architectural landscape.

```
slicesupportlist:
sst: "01"
sd: "000001"
```

Network Slicing



AF= Application functions

UPF= User plane functions

DN= Data Network

AMF=Access and mobility management Function

SMF=Session management function

Fig 2: Network Slicing

PacketRusher

- ✓ PacketRusher, a cutting-edge 5G core Network performance testing tool.
- ✓ Its primary objective is to establish automated and high-performance UE/RAN testing suites.
- ✓ Packet Rusher itself is in a developing stage however the configuration seems similar to UERANSIM

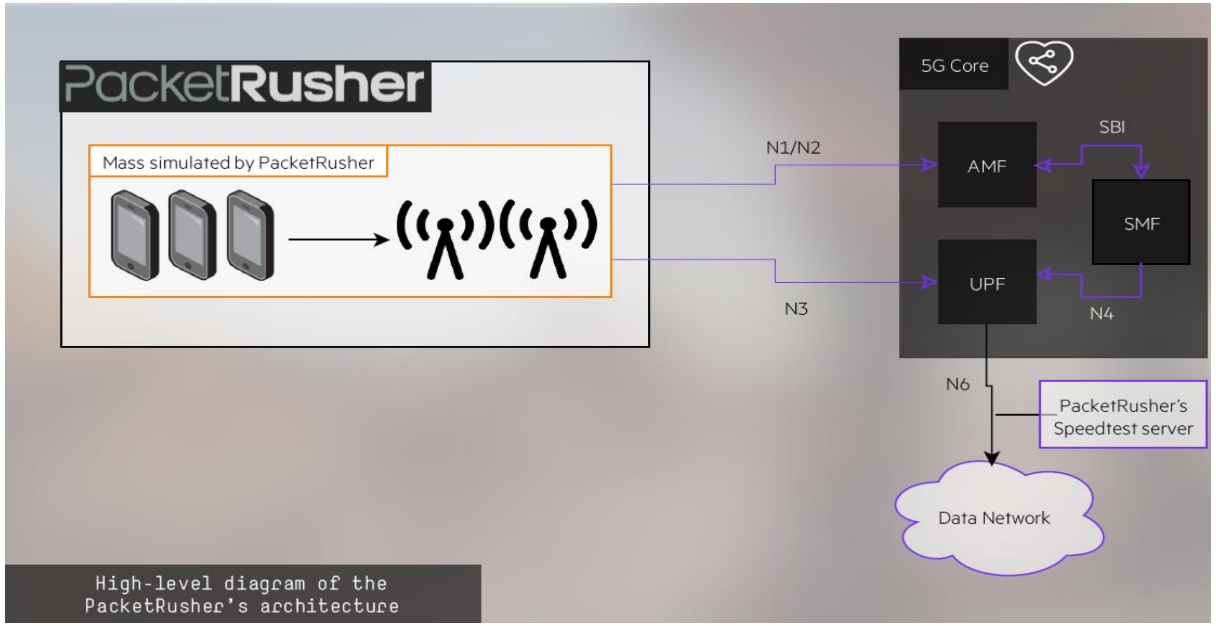

PacketRusher
Mass simulated by PacketRusher
N1/N2
N3
5G Core
SBI
AMF
SMF
UPF
N4
N6
PacketRusher's Speedtest server
Data Network
High-level diagram of the PacketRusher's architecture

Fig 4: PacketRusher

Image Source

Common Feature of PacketRusher

- Simulate multiple UEs and gNodeB from a single tool
- Supports both N2 (NGAP) and N1 (NAS) interfaces for stress testing
- pcap parameter to capture pcap of N1/N2 traffic
- Implements main control plane
- Implements high-performant N3 (GTP-U) interface
- Integrated all-in-one mocked 5GC/AMF for PacketRusher's integration testing

RAN, GNB & UEs

RAN:

Basically, RAN in 5G consists of:

- gNB (Next-Generation NodeB) - 5G base station that handles signal processing and network access.
- CU (Centralized Unit) and DU (Distributed Unit) – Splitting network processing for efficiency and scalability.
- RU (Radio Unit) – Handles radio transmission at the antenna level.

UEs:

Devices including smartphones, Internet of Things devices, and industrial sensors that connect to the 5G network are referred to as UEs. To create network connectivity, these devices speak with the gNB.

System Architecture

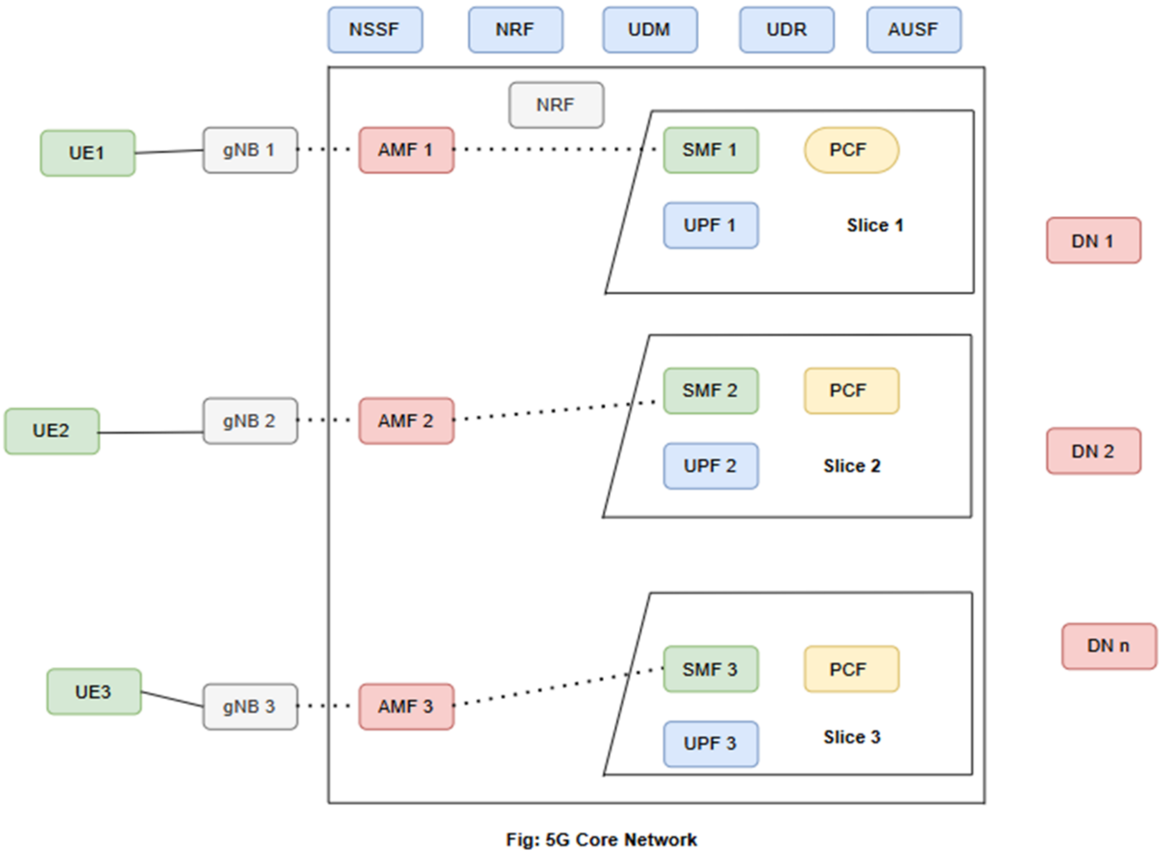
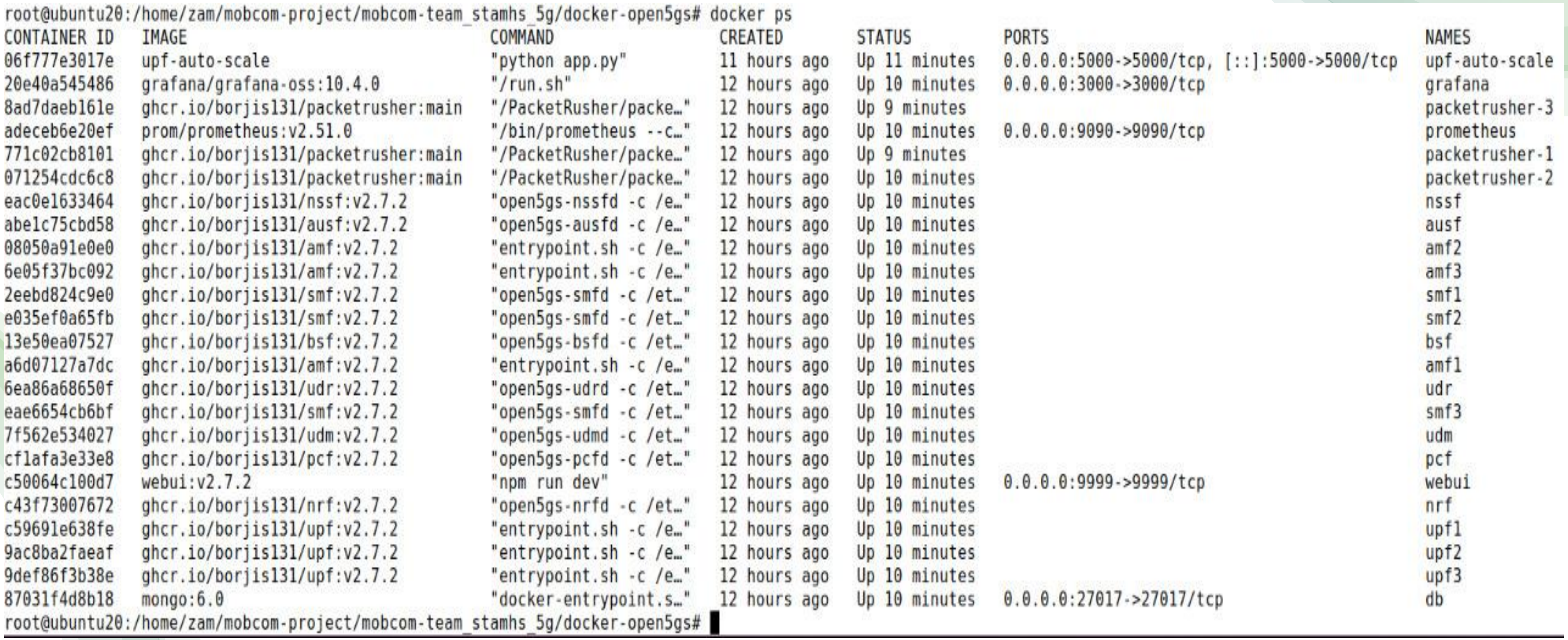


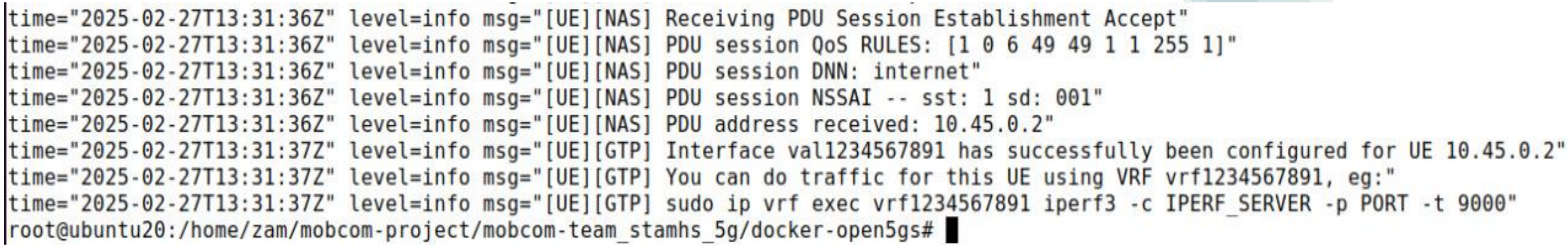
Fig 5: System Architecture

Containerized Deployment Overview

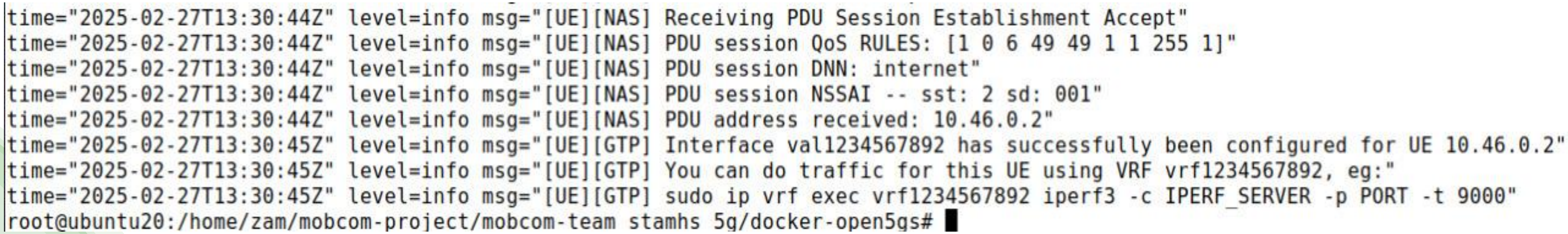


```
root@ubuntu20:/home/zam/mobcom-project/mobcom-team_stamhs_5g/docker-open5gs# docker ps
CONTAINER ID   IMAGE                                 COMMAND                  CREATED        STATUS          PORTS                                          NAMES
06f777e3017e   upf-auto-scale                        "python app.py"          11 hours ago   Up 11 minutes   0.0.0.0:5000->5000/tcp, [::]:5000->5000/tcp   upf-auto-scale
20e40a545486   grafana/grafana-oss:10.4.0            "/run.sh"                12 hours ago   Up 10 minutes   0.0.0.0:3000->3000/tcp                         grafana
8ad7daeb161e   ghcr.io/borjis131/packetrusher:main   "/PacketRusher/packe…"   12 hours ago   Up 9 minutes                                                   packetrusher-3
adeceb6e20ef   prom/prometheus:v2.51.0               "/bin/prometheus --c…"   12 hours ago   Up 10 minutes   0.0.0.0:9090->9090/tcp                         prometheus
771c02cb8101   ghcr.io/borjis131/packetrusher:main   "/PacketRusher/packe…"   12 hours ago   Up 9 minutes                                                   packetrusher-1
071254cdc6c8   ghcr.io/borjis131/packetrusher:main   "/PacketRusher/packe…"   12 hours ago   Up 10 minutes                                                  packetrusher-2
eac0e1633464   ghcr.io/borjis131/nssf:v2.7.2         "open5gs-nssfd -c /e…"   12 hours ago   Up 10 minutes                                                  nssf
abe1c75cbd58   ghcr.io/borjis131/ausf:v2.7.2         "open5gs-ausfd -c /e…"   12 hours ago   Up 10 minutes                                                  ausf
08050a91e0e0   ghcr.io/borjis131/amf:v2.7.2          "entrypoint.sh -c /e…"   12 hours ago   Up 10 minutes                                                  amf2
6e05f37bc092   ghcr.io/borjis131/amf:v2.7.2          "entrypoint.sh -c /e…"   12 hours ago   Up 10 minutes                                                  amf3
2eebd824c9e0   ghcr.io/borjis131/smf:v2.7.2          "open5gs-smfd -c /et…"   12 hours ago   Up 10 minutes                                                  smf1
e035ef0a65fb   ghcr.io/borjis131/smf:v2.7.2          "open5gs-smfd -c /et…"   12 hours ago   Up 10 minutes                                                  smf2
13e50ea07527   ghcr.io/borjis131/bsf:v2.7.2          "open5gs-bsfd -c /et…"   12 hours ago   Up 10 minutes                                                  bsf
a6d07127a7dc   ghcr.io/borjis131/amf:v2.7.2          "entrypoint.sh -c /e…"   12 hours ago   Up 10 minutes                                                  amf1
6ea86a68650f   ghcr.io/borjis131/udr:v2.7.2          "open5gs-udrd -c /et…"   12 hours ago   Up 10 minutes                                                  udr
eae6654cb6bf   ghcr.io/borjis131/smf:v2.7.2          "open5gs-smfd -c /et…"   12 hours ago   Up 10 minutes                                                  smf3
7f562e534027   ghcr.io/borjis131/udm:v2.7.2          "open5gs-udmd -c /et…"   12 hours ago   Up 10 minutes                                                  udm
cf1afa3e33e8   ghcr.io/borjis131/pcf:v2.7.2          "open5gs-pcfd -c /et…"   12 hours ago   Up 10 minutes                                                  pcf
c50064c100d7   webui:v2.7.2                          "npm run dev"            12 hours ago   Up 10 minutes   0.0.0.0:9999->9999/tcp                         webui
c43f73007672   ghcr.io/borjis131/nrf:v2.7.2          "open5gs-nrfd -c /et…"   12 hours ago   Up 10 minutes                                                  nrf
c59691e638fe   ghcr.io/borjis131/upf:v2.7.2          "entrypoint.sh -c /e…"   12 hours ago   Up 10 minutes                                                  upf1
9ac8ba2faeaf   ghcr.io/borjis131/upf:v2.7.2          "entrypoint.sh -c /e…"   12 hours ago   Up 10 minutes                                                  upf2
9def86f3b38e   ghcr.io/borjis131/upf:v2.7.2          "entrypoint.sh -c /e…"   12 hours ago   Up 10 minutes                                                  upf3
87031f4d8b18   mongo:6.0                             "docker-entrypoint.s…"   12 hours ago   Up 10 minutes   0.0.0.0:27017->27017/tcp                       db
root@ubuntu20:/home/zam/mobcom-project/mobcom-team_stamhs_5g/docker-open5gs# 
```

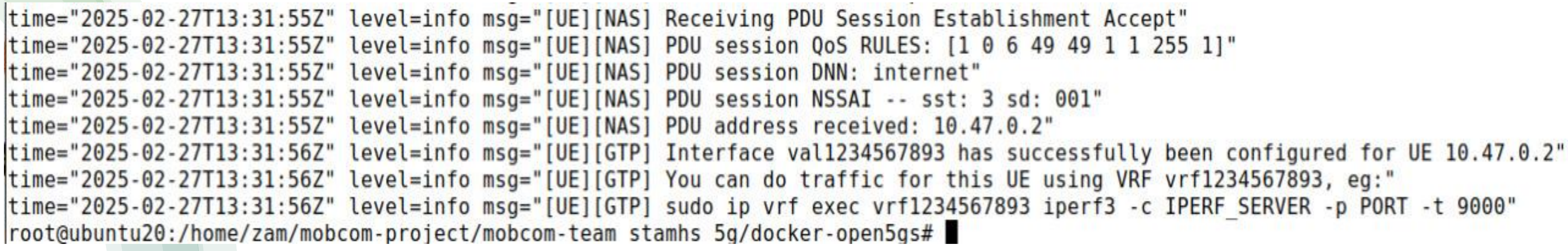
UE Registration & Session Establishment



```
time="2025-02-27T13:31:36Z" level=info msg="[UE][NAS] Receiving PDU Session Establishment Accept"
time="2025-02-27T13:31:36Z" level=info msg="[UE][NAS] PDU session QoS RULES: [1 0 6 49 49 1 1 255 1]"
time="2025-02-27T13:31:36Z" level=info msg="[UE][NAS] PDU session DNN: internet"
time="2025-02-27T13:31:36Z" level=info msg="[UE][NAS] PDU session NSSAI -- sst: 1 sd: 001"
time="2025-02-27T13:31:36Z" level=info msg="[UE][NAS] PDU address received: 10.45.0.2"
time="2025-02-27T13:31:37Z" level=info msg="[UE][GTP] Interface val1234567891 has successfully been configured for UE 10.45.0.2"
time="2025-02-27T13:31:37Z" level=info msg="[UE][GTP] You can do traffic for this UE using VRF vrf1234567891, eg:"
time="2025-02-27T13:31:37Z" level=info msg="[UE][GTP] sudo ip vrf exec vrf1234567891 iperf3 -c IPERF_SERVER -p PORT -t 9000"
root@ubuntu20:/home/zam/mobcom-project/mobcom-team_stamhs_5g/docker-open5gs#
```

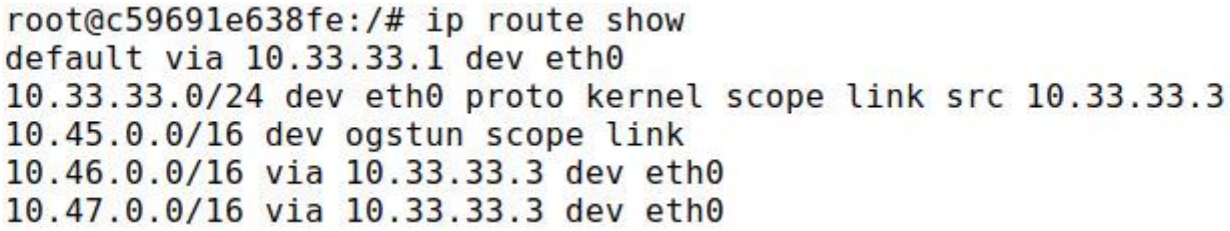


```
time="2025-02-27T13:30:44Z" level=info msg="[UE][NAS] Receiving PDU Session Establishment Accept"
time="2025-02-27T13:30:44Z" level=info msg="[UE][NAS] PDU session QoS RULES: [1 0 6 49 49 1 1 255 1]"
time="2025-02-27T13:30:44Z" level=info msg="[UE][NAS] PDU session DNN: internet"
time="2025-02-27T13:30:44Z" level=info msg="[UE][NAS] PDU session NSSAI -- sst: 2 sd: 001"
time="2025-02-27T13:30:44Z" level=info msg="[UE][NAS] PDU address received: 10.46.0.2"
time="2025-02-27T13:30:45Z" level=info msg="[UE][GTP] Interface val1234567892 has successfully been configured for UE 10.46.0.2"
time="2025-02-27T13:30:45Z" level=info msg="[UE][GTP] You can do traffic for this UE using VRF vrf1234567892, eg:"
time="2025-02-27T13:30:45Z" level=info msg="[UE][GTP] sudo ip vrf exec vrf1234567892 iperf3 -c IPERF_SERVER -p PORT -t 9000"
root@ubuntu20:/home/zam/mobcom-project/mobcom-team stamhs 5g/docker-open5gs#
```

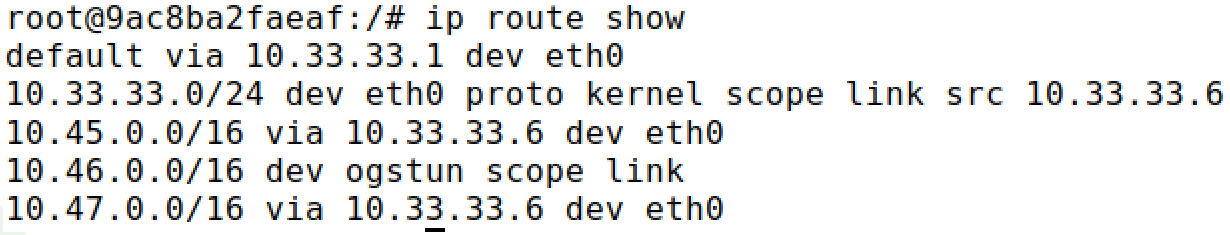


```
time="2025-02-27T13:31:55Z" level=info msg="[UE][NAS] Receiving PDU Session Establishment Accept"
time="2025-02-27T13:31:55Z" level=info msg="[UE][NAS] PDU session QoS RULES: [1 0 6 49 49 1 1 255 1]"
time="2025-02-27T13:31:55Z" level=info msg="[UE][NAS] PDU session DNN: internet"
time="2025-02-27T13:31:55Z" level=info msg="[UE][NAS] PDU session NSSAI -- sst: 3 sd: 001"
time="2025-02-27T13:31:55Z" level=info msg="[UE][NAS] PDU address received: 10.47.0.2"
time="2025-02-27T13:31:56Z" level=info msg="[UE][GTP] Interface val1234567893 has successfully been configured for UE 10.47.0.2"
time="2025-02-27T13:31:56Z" level=info msg="[UE][GTP] You can do traffic for this UE using VRF vrf1234567893, eg:"
time="2025-02-27T13:31:56Z" level=info msg="[UE][GTP] sudo ip vrf exec vrf1234567893 iperf3 -c IPERF_SERVER -p PORT -t 9000"
root@ubuntu20:/home/zam/mobcom-project/mobcom-team stamhs 5g/docker-open5gs#
```

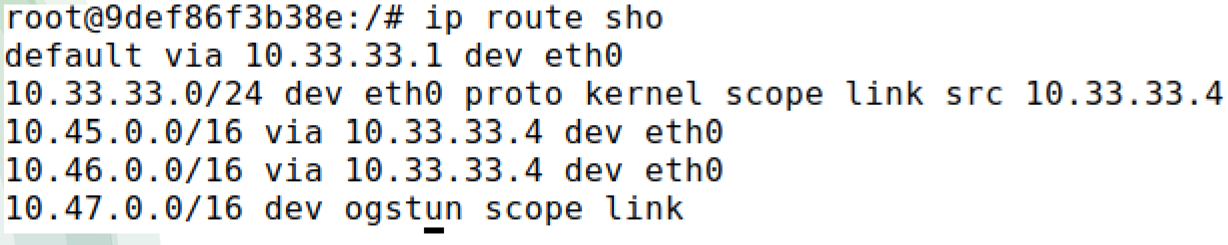
IP configurations of UPFs and Packetrusher



```
root@c59691e638fe:/# ip route show
default via 10.33.33.1 dev eth0
10.33.33.0/24 dev eth0 proto kernel scope link src 10.33.33.3
10.45.0.0/16 dev ogstun scope link
10.46.0.0/16 via 10.33.33.3 dev eth0
10.47.0.0/16 via 10.33.33.3 dev eth0
```



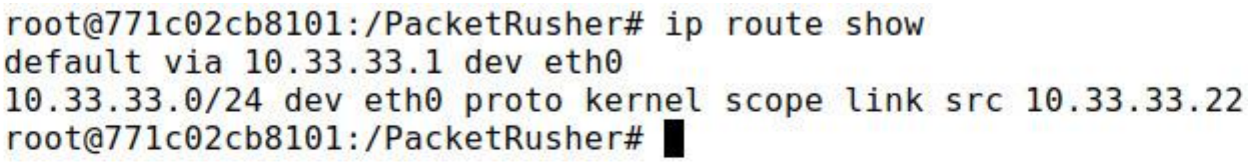
```
root@9ac8ba2faeaf:/# ip route show
default via 10.33.33.1 dev eth0
10.33.33.0/24 dev eth0 proto kernel scope link src 10.33.33.6
10.45.0.0/16 via 10.33.33.6 dev eth0
10.46.0.0/16 dev ogstun scope link
10.47.0.0/16 via 10.33.33.6 dev eth0
```



```
root@9def86f3b38e:/# ip route sho
default via 10.33.33.1 dev eth0
10.33.33.0/24 dev eth0 proto kernel scope link src 10.33.33.4
10.45.0.0/16 via 10.33.33.4 dev eth0
10.46.0.0/16 via 10.33.33.4 dev eth0
10.47.0.0/16 dev ogstun scope link
```

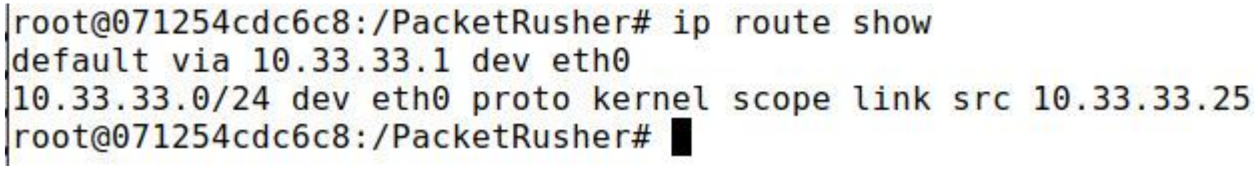
IP configurations of UPFs and Packetrusher

Packetrusher -1 (UE1 Traffic)



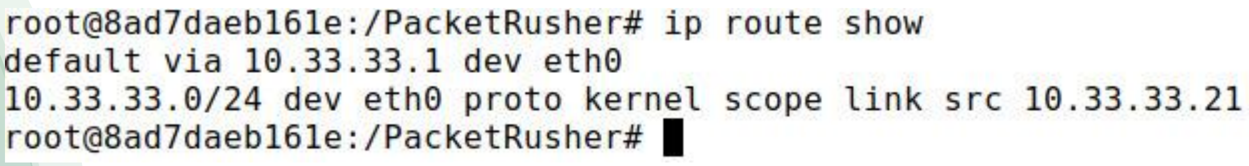
```
root@771c02cb8101:/PacketRusher# ip route show
default via 10.33.33.1 dev eth0
10.33.33.0/24 dev eth0 proto kernel scope link src 10.33.33.22
root@771c02cb8101:/PacketRusher# █
```

Packetrusher -2 (UE2 Traffic)



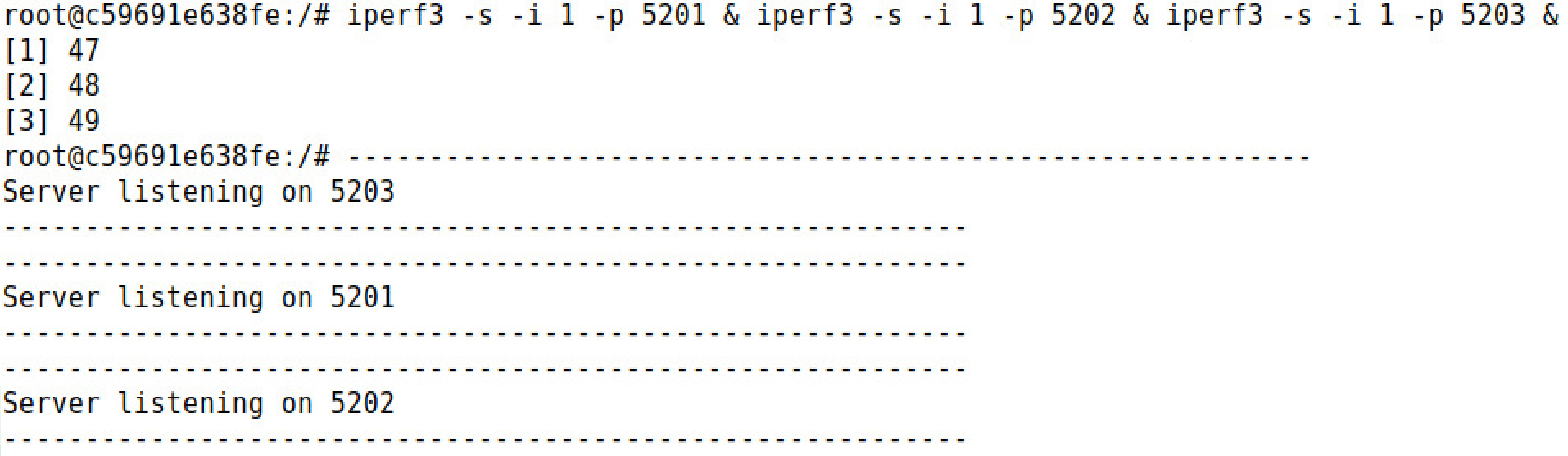
```
root@071254cdc6c8:/PacketRusher# ip route show
default via 10.33.33.1 dev eth0
10.33.33.0/24 dev eth0 proto kernel scope link src 10.33.33.25
root@071254cdc6c8:/PacketRusher# █
```

Packetrusher -3 (UE3 Traffic)



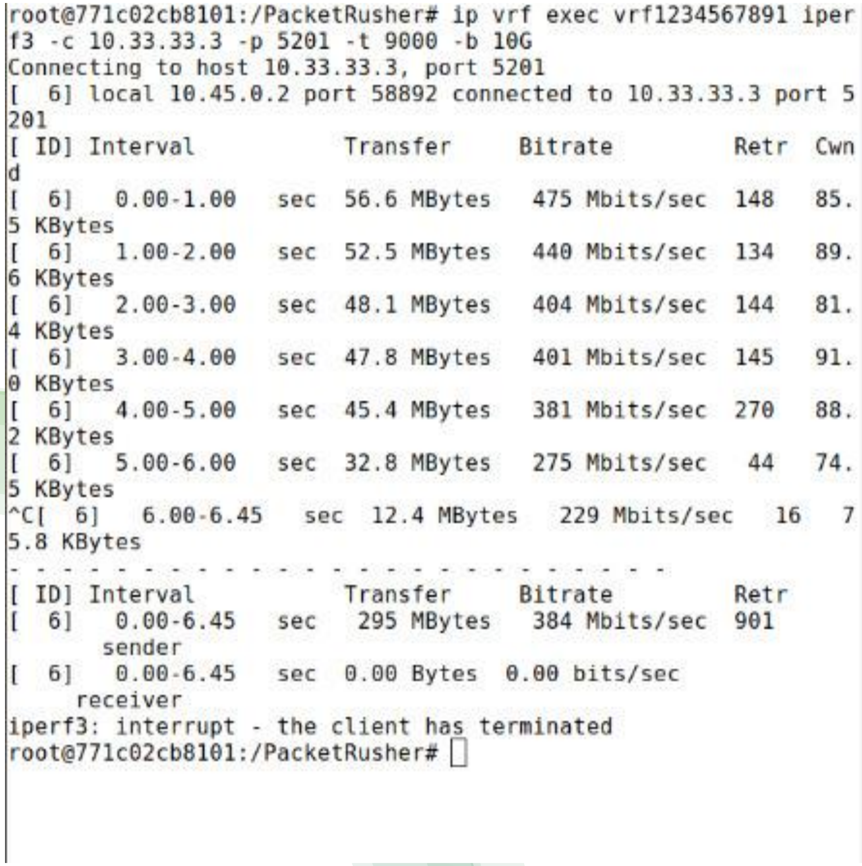
```
root@8ad7daeb161e:/PacketRusher# ip route show
default via 10.33.33.1 dev eth0
10.33.33.0/24 dev eth0 proto kernel scope link src 10.33.33.21
root@8ad7daeb161e:/PacketRusher# █
```

Listening to receive traffic

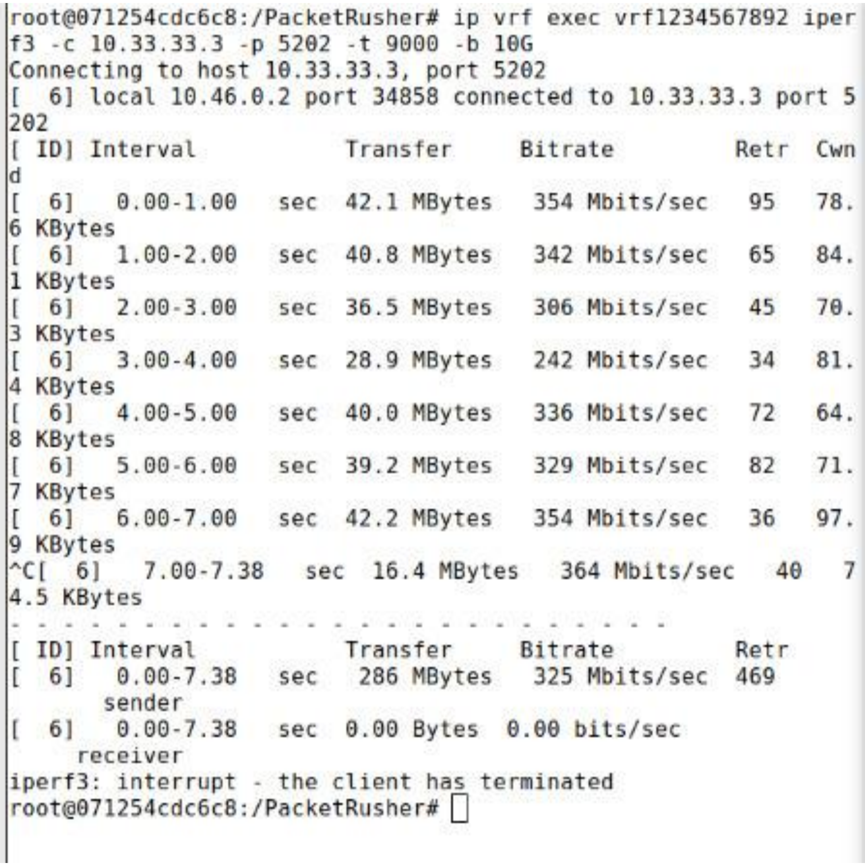


```
root@c59691e638fe:/# iperf3 -s -i 1 -p 5201 & iperf3 -s -i 1 -p 5202 & iperf3 -s -i 1 -p 5203 &
[1] 47
[2] 48
[3] 49
root@c59691e638fe:/# -----------------------------------------------------------
Server listening on 5203
-----------------------------------------------------------
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
-----------------------------------------------------------
Server listening on 5202
-----------------------------------------------------------
```

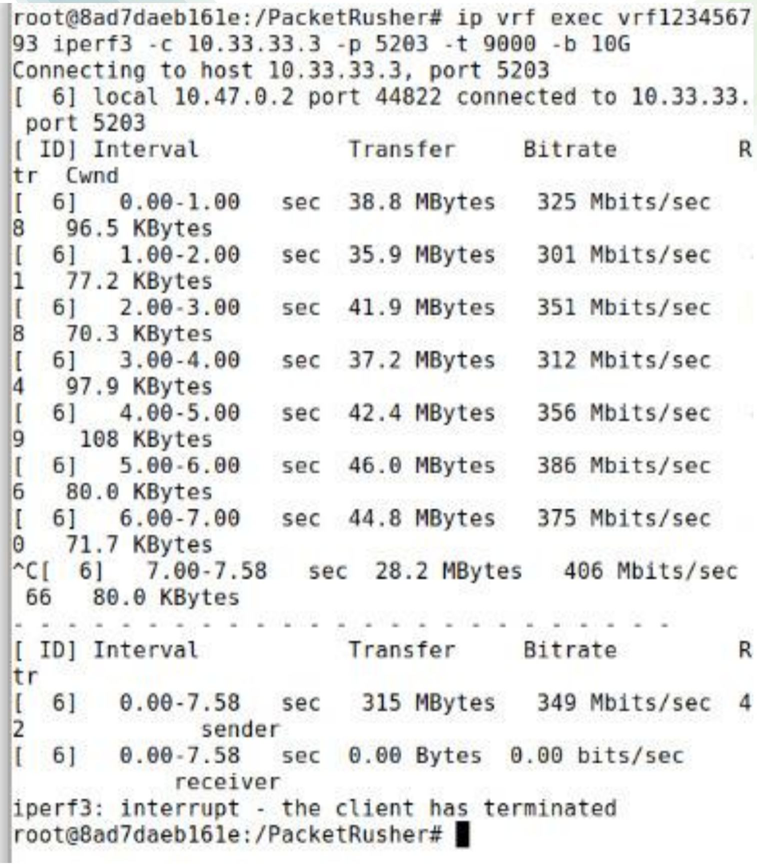
Traffic Throughput for Each UE



```
root@771c02cb8101:/PacketRusher# ip vrf exec vrf1234567891 iper
f3 -c 10.33.33.3 -p 5201 -t 9000 -b 10G
Connecting to host 10.33.33.3, port 5201
[  6] local 10.45.0.2 port 58892 connected to 10.33.33.3 port 5
201
[ ID] Interval           Transfer     Bitrate         Retr  Cwn
d
[  6]   0.00-1.00   sec  56.6 MBytes   475 Mbits/sec  148   85.
5 KBytes
[  6]   1.00-2.00   sec  52.5 MBytes   440 Mbits/sec  134   89.
6 KBytes
[  6]   2.00-3.00   sec  48.1 MBytes   404 Mbits/sec  144   81.
4 KBytes
[  6]   3.00-4.00   sec  47.8 MBytes   401 Mbits/sec  145   91.
0 KBytes
[  6]   4.00-5.00   sec  45.4 MBytes   381 Mbits/sec  270   88.
2 KBytes
[  6]   5.00-6.00   sec  32.8 MBytes   275 Mbits/sec   44   74.
5 KBytes
^C[  6]   6.00-6.45   sec  12.4 MBytes   229 Mbits/sec   16   7
5.8 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  6]   0.00-6.45   sec   295 MBytes   384 Mbits/sec  901
        sender
[  6]   0.00-6.45   sec  0.00 Bytes  0.00 bits/sec
      receiver
iperf3: interrupt - the client has terminated
root@771c02cb8101:/PacketRusher# 
```

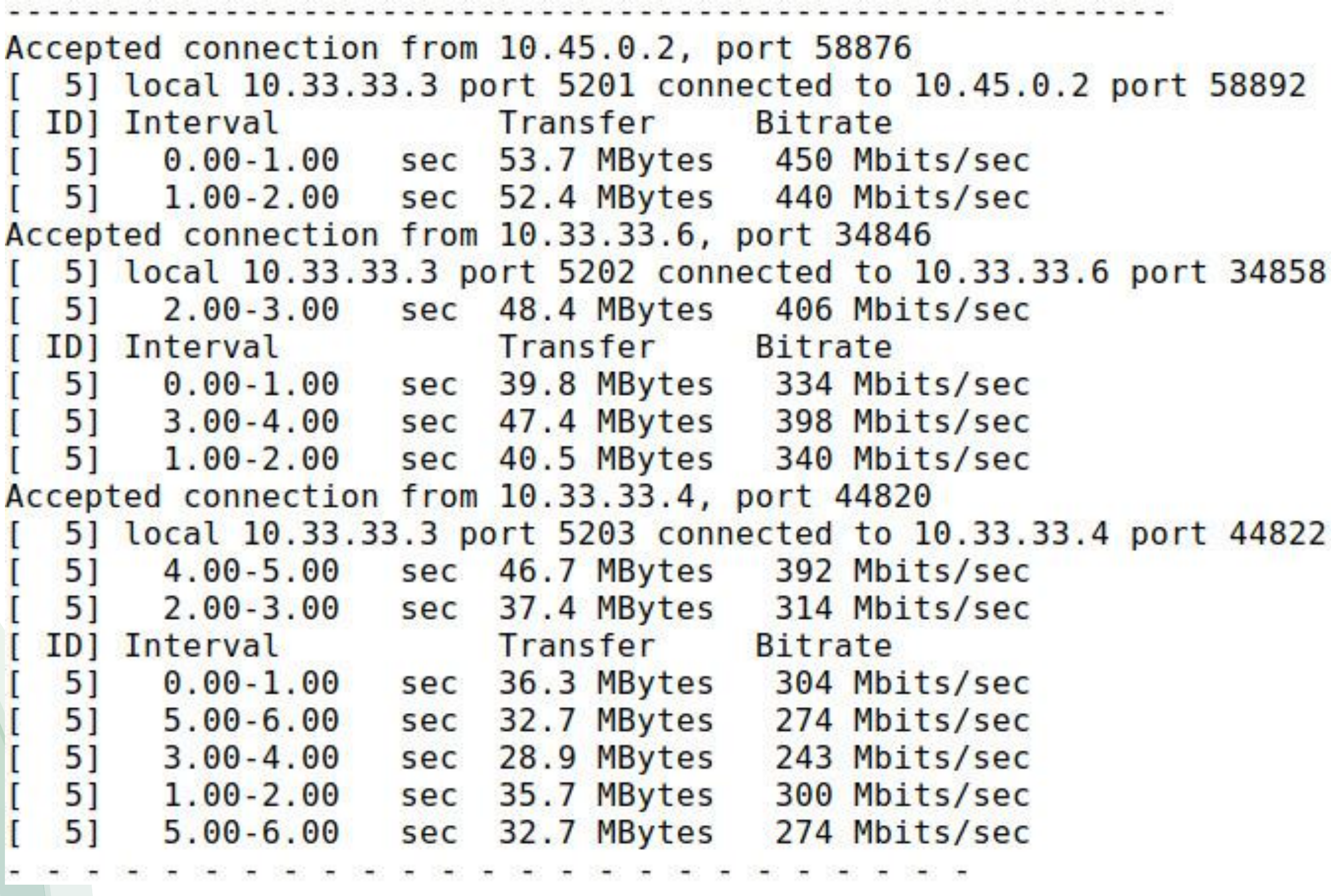


```
root@071254cdc6c8:/PacketRusher# ip vrf exec vrf1234567892 iper
f3 -c 10.33.33.3 -p 5202 -t 9000 -b 10G
Connecting to host 10.33.33.3, port 5202
[  6] local 10.46.0.2 port 34858 connected to 10.33.33.3 port 5
202
[ ID] Interval           Transfer     Bitrate         Retr  Cwn
d
[  6]   0.00-1.00   sec  42.1 MBytes   354 Mbits/sec   95   78.
6 KBytes
[  6]   1.00-2.00   sec  40.8 MBytes   342 Mbits/sec   65   84.
1 KBytes
[  6]   2.00-3.00   sec  36.5 MBytes   306 Mbits/sec   45   70.
3 KBytes
[  6]   3.00-4.00   sec  28.9 MBytes   242 Mbits/sec   34   81.
4 KBytes
[  6]   4.00-5.00   sec  40.0 MBytes   336 Mbits/sec   72   64.
8 KBytes
[  6]   5.00-6.00   sec  39.2 MBytes   329 Mbits/sec   82   71.
7 KBytes
[  6]   6.00-7.00   sec  42.2 MBytes   354 Mbits/sec   36   97.
9 KBytes
^C[  6]   7.00-7.38   sec  16.4 MBytes   364 Mbits/sec   40   7
4.5 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  6]   0.00-7.38   sec   286 MBytes   325 Mbits/sec  469
        sender
[  6]   0.00-7.38   sec  0.00 Bytes  0.00 bits/sec
      receiver
iperf3: interrupt - the client has terminated
root@071254cdc6c8:/PacketRusher# 
```



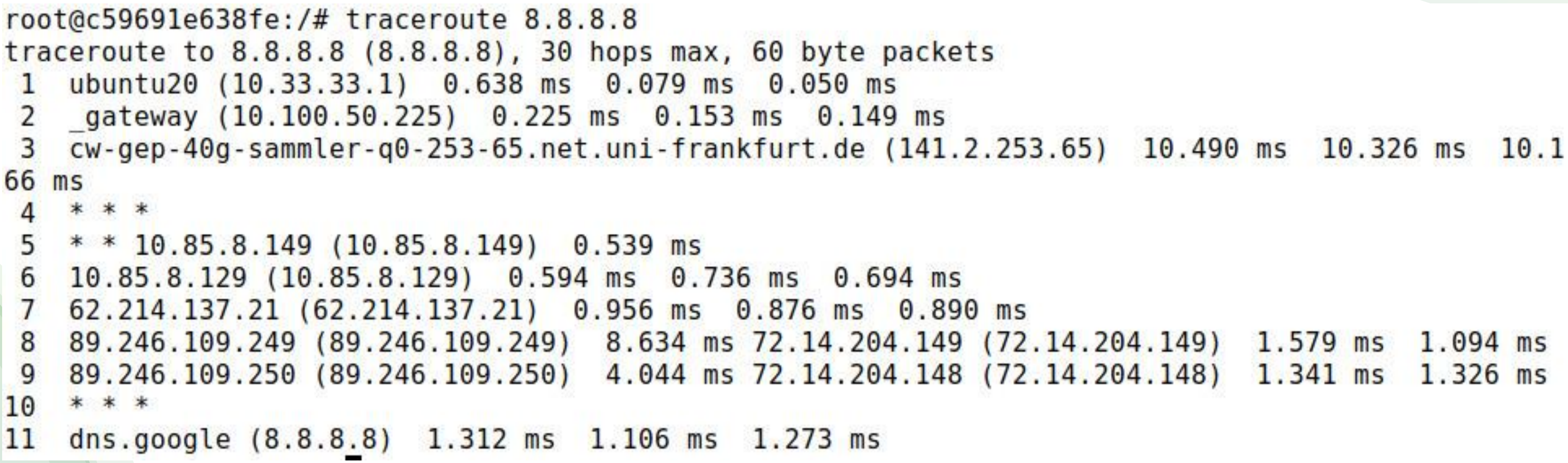
```
root@8ad7daeb161e:/PacketRusher# ip vrf exec vrf1234567
93 iperf3 -c 10.33.33.3 -p 5203 -t 9000 -b 10G
Connecting to host 10.33.33.3, port 5203
[  6] local 10.47.0.2 port 44822 connected to 10.33.33.
 port 5203
[ ID] Interval           Transfer     Bitrate         R
tr  Cwnd
[  6]   0.00-1.00   sec  38.8 MBytes   325 Mbits/sec
8   96.5 KBytes
[  6]   1.00-2.00   sec  35.9 MBytes   301 Mbits/sec
1   77.2 KBytes
[  6]   2.00-3.00   sec  41.9 MBytes   351 Mbits/sec
8   70.3 KBytes
[  6]   3.00-4.00   sec  37.2 MBytes   312 Mbits/sec
4   97.9 KBytes
[  6]   4.00-5.00   sec  42.4 MBytes   356 Mbits/sec
9    108 KBytes
[  6]   5.00-6.00   sec  46.0 MBytes   386 Mbits/sec
6   80.0 KBytes
[  6]   6.00-7.00   sec  44.8 MBytes   375 Mbits/sec
0   71.7 KBytes
^C[  6]   7.00-7.58   sec  28.2 MBytes   406 Mbits/sec
 66   80.0 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         R
tr
[  6]   0.00-7.58   sec   315 MBytes   349 Mbits/sec  4
2             sender
[  6]   0.00-7.58   sec  0.00 Bytes  0.00 bits/sec
            receiver
iperf3: interrupt - the client has terminated
root@8ad7daeb161e:/PacketRusher# 
```

Traffic Received at UPF



```
------------------------------------------------------------
Accepted connection from 10.45.0.2, port 58876
[  5] local 10.33.33.3 port 5201 connected to 10.45.0.2 port 58892
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.00   sec  53.7 MBytes   450 Mbits/sec
[  5]   1.00-2.00   sec  52.4 MBytes   440 Mbits/sec
Accepted connection from 10.33.33.6, port 34846
[  5] local 10.33.33.3 port 5202 connected to 10.33.33.6 port 34858
[  5]   2.00-3.00   sec  48.4 MBytes   406 Mbits/sec
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.00   sec  39.8 MBytes   334 Mbits/sec
[  5]   3.00-4.00   sec  47.4 MBytes   398 Mbits/sec
[  5]   1.00-2.00   sec  40.5 MBytes   340 Mbits/sec
Accepted connection from 10.33.33.4, port 44820
[  5] local 10.33.33.3 port 5203 connected to 10.33.33.4 port 44822
[  5]   4.00-5.00   sec  46.7 MBytes   392 Mbits/sec
[  5]   2.00-3.00   sec  37.4 MBytes   314 Mbits/sec
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.00   sec  36.3 MBytes   304 Mbits/sec
[  5]   5.00-6.00   sec  32.7 MBytes   274 Mbits/sec
[  5]   3.00-4.00   sec  28.9 MBytes   243 Mbits/sec
[  5]   1.00-2.00   sec  35.7 MBytes   300 Mbits/sec
[  5]   5.00-6.00   sec  32.7 MBytes   274 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
```

UPF Internet Connectivity Test



```
root@c59691e638fe:/# traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  ubuntu20 (10.33.33.1)  0.638 ms  0.079 ms  0.050 ms
 2  _gateway (10.100.50.225)  0.225 ms  0.153 ms  0.149 ms
 3  cw-gep-40g-sammler-q0-253-65.net.uni-frankfurt.de (141.2.253.65)  10.490 ms  10.326 ms  10.1
66 ms
 4  * * *
 5  * * 10.85.8.149 (10.85.8.149)  0.539 ms
 6  10.85.8.129 (10.85.8.129)  0.594 ms  0.736 ms  0.694 ms
 7  62.214.137.21 (62.214.137.21)  0.956 ms  0.876 ms  0.890 ms
 8  89.246.109.249 (89.246.109.249)  8.634 ms 72.14.204.149 (72.14.204.149)  1.579 ms  1.094 ms
 9  89.246.109.250 (89.246.109.250)  4.044 ms 72.14.204.148 (72.14.204.148)  1.341 ms  1.326 ms
10  * * *
11  dns.google (8.8.8.8)  1.312 ms  1.106 ms  1.273 ms
```

FRANKFURT
UNIVERSITY
OF APPLIED SCIENCES

Challenges and Limitations

- Unable to integrate Multiple UEs under single gNB due to PacketRusher’s limitations.
- Auto scale up of UPFs during traffic overload couldn’t be implemented.

Conclusion

In this project we have created a Standalone 5G Core Network (5GC) with Open5GS featuring support for network slicing, multiple User Plane Functions (UPFs), and a containerized architecture. This architecture efficiently routes traffic among UPFs, User Equipment (UEs), and gNodeBs (gNBs) efficiently. We have created an efficient 5G RAN simulation with PacketRusher support featuring multiple gNBs and UEs to support extensive traffic generation and testing. Besides this, iperf3 testing checks that traffic between different UEs is correctly divided between UPFs in accordance with S-NSSAI to ensure proper network slicing.

References

[1] U. Trick, 5G: The 5th Generation Mobile Networks, 2nd ed.

[2] https://github.com/HewlettPackard/PacketRusher

[3] https://info.support.huawei.com/info-finder/encyclopedia/en/Network+Slicing.html

[4] https://www.viavisolutions.com/en-us/5g-network-slicing

[5] https://developer.hpe.com/blog/open-sourcing-packetrusher-a-5g-core-performance-tester/

Thank You