Magma Glacier Peak (v1.6) - Validation

Setup

Access Gateway Hardware

Physical Gateways

- CPU - Intel(R) Celeron(R) CPU J3160 1.6Ghz
- Memory - 8GB DDR3 1066Mhz
- Storage - 120G SSD
- Network - 4x1 Gigabit Network Connection (rev03)

Traffic Emulator

- Spirent Landslide UE and eNB emulator with 1G traffic ports

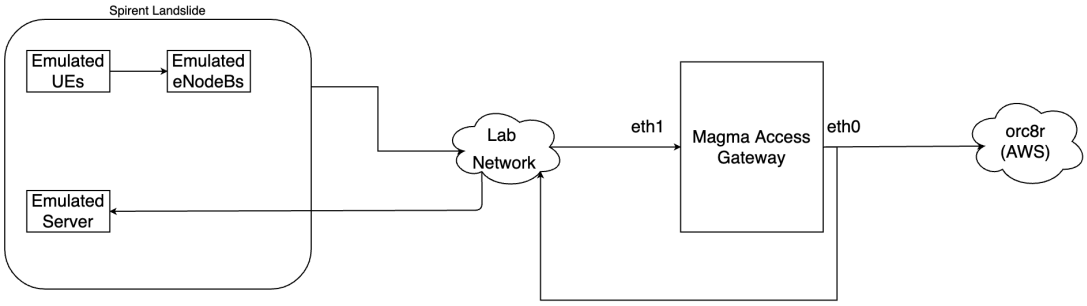


Fig 1 - High level block block diagram for the emulated setup for a NAT setup

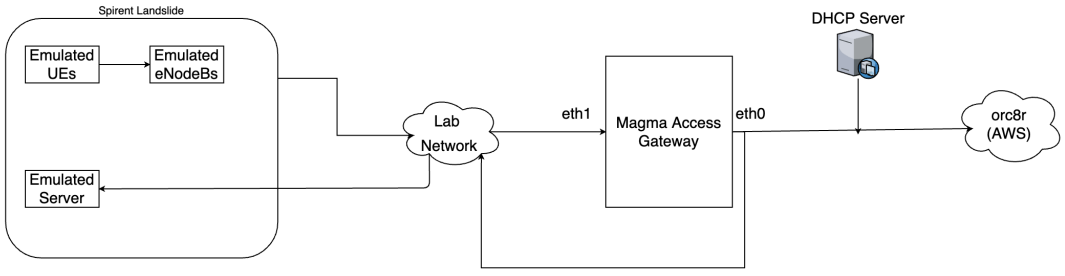


Fig 2 - High level block diagram for non-nat DHCP setup

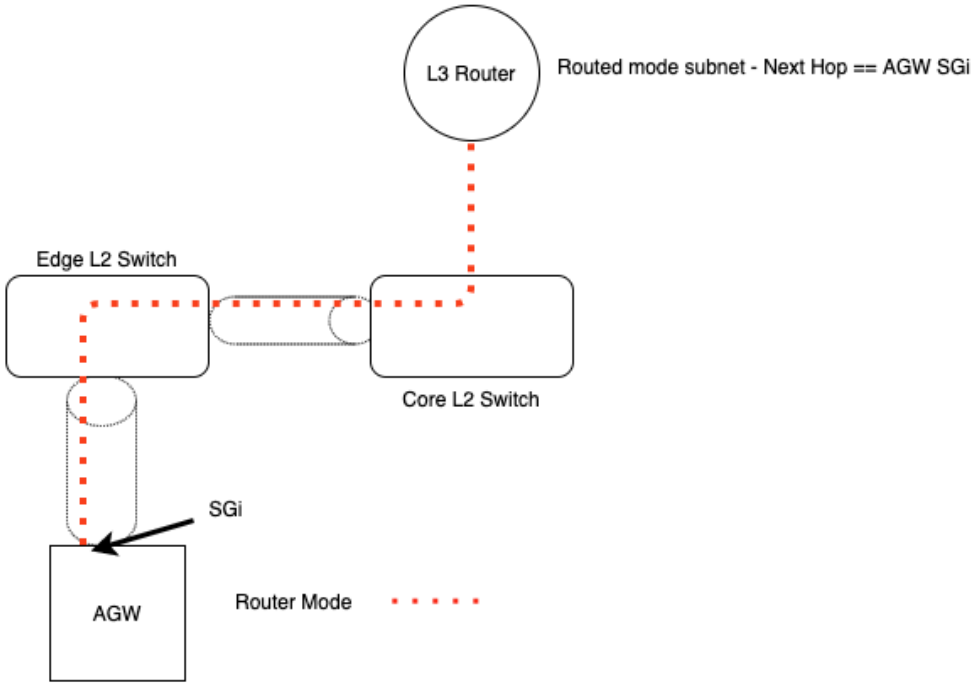
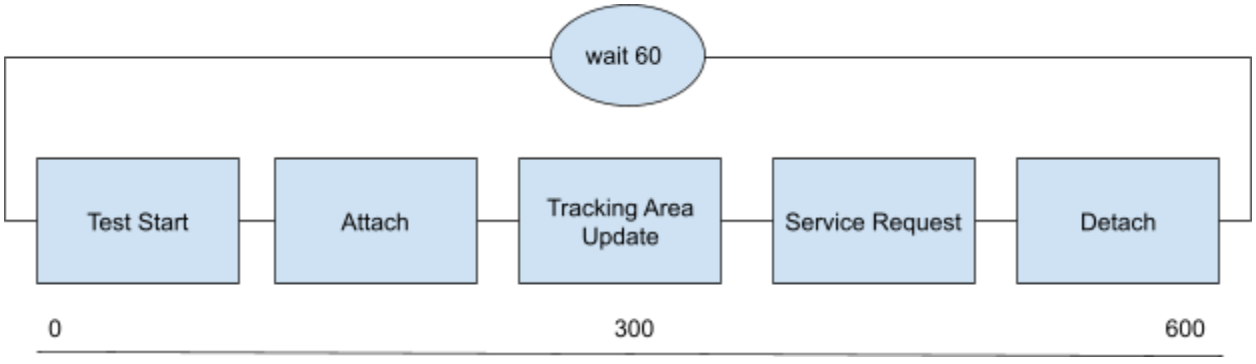


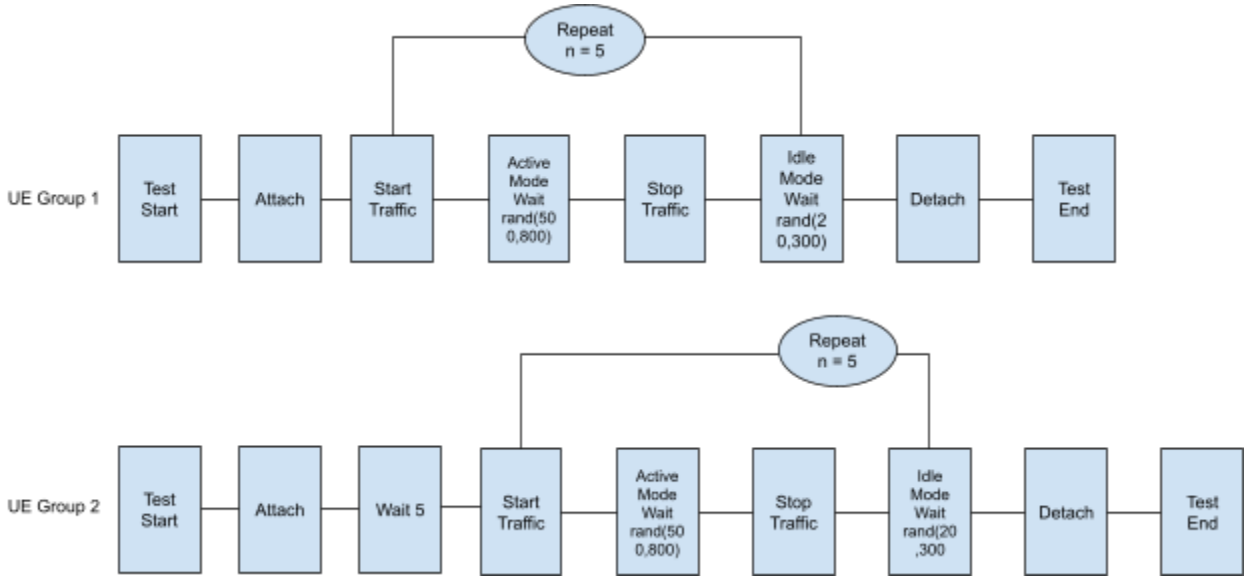
Fig 3 - High level block diagram showing the router mode setup.

Test Activity

- Attach & Detach tests



- Active/Idle mode tests (Inbound Roaming)


Repeat
n = 5
UE Group 1
Test Start
Attach
Start Traffic
Active Mode Wait rand(50 0,800)
Stop Traffic
Idle Mode Wait rand(2 0,300)
Detach
Test End
Repeat
n = 5
UE Group 2
Test Start
Attach
Wait 5
Start Traffic
Active Mode Wait rand(50 0,800)
Stop Traffic
Idle Mode Wait rand(20 ,300
Detach
Test End

Install, Upgrade Validation

AGW BareMetal

Ubuntu 1.6.0 Fresh Install

Download and run
`wget https://raw.githubusercontent.com/magma/magma/master/lte/gateway/deploy/agw_install_ubuntu.sh`
Install Logs

Network Type Tested

- NAT
- non-NAT with DHCP (SGi)
- non-NAT with Static IP (SGi)

AGW Upgrade

Download and run
`wget https://raw.githubusercontent.com/magma/magma/master/lte/gateway/release/upgrade_magma.sh`

Ubuntu	1.6	Upgrade Logs & Verification
1.5.0		Install Logs
1.5.1 • NAT • Bridge Mode	• NAT • Bridge Mode	Install Logs
1.5.2 • NAT • Bridge Mode	• NAT • Bridge Mode	Install Logs

Orc8r

v1.6 Fresh Install (Cloudstrapper)

- Create NMS Users
    - Logout/Login
- Create Subscribers
- Checkin Gateway
- Sync predefined alerts
- Check Metrics

Cloudstrapper logs

v1.6 Fresh Install (Terraform)

Same images as cloudstrapper were used hence no additional validation other than NMS access and kubectl command line access.

Terraform install logs

v1.5 -> 1.6 Upgrade (Terraform)

- Create NMS Users
    - Logout/Login
- Create Subscribers
- Checkin Gateway
- Sync predefined alerts
- Check Metrics

Terraform upgrade logs

Cloudstrapper general notes:

- `TASK [control : inside terraform home run terraform apply for orc8r]` - This step takes quite a while to finish. The user will probably not see anything on the screen while this step is going on.
- If Terraform times out during the `terraform apply --target=module.orc8r`, the recommended next step is to investigate the cause of the timeout. Depending on if all infra has been setup (VPC, Subnets, Security Rules, EKS cluster, ES cluster, EFS, RDS etc.) the user can cautiously proceed to applying the rest of the terraform commands manually; or see the
- If a Cloudstrapper run partially finished (i.e. some changes were made in AWS), it is recommended to run a cleanup script before proceeding with another install. Furthermore, if a cleanup script is run, the corresponding tfstate files need to be manually deleted before proceeding with the next installation attempt.

Feature Testing

AGW

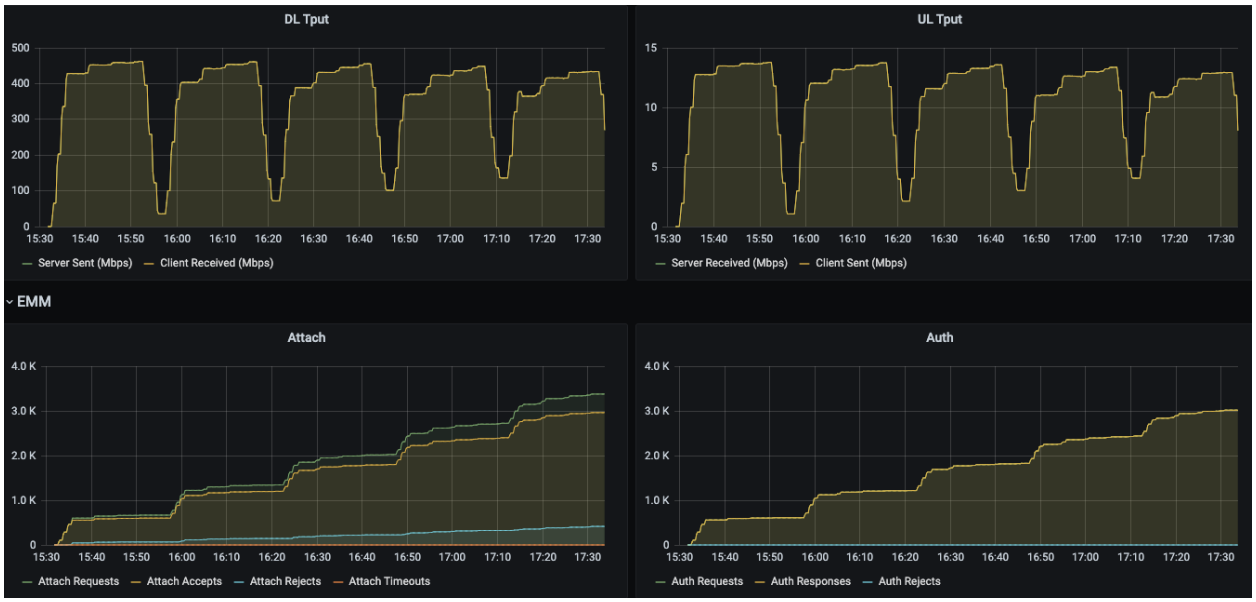
Multi APN with Static IP + DHCP

Ubuntu -

- eNBs 12
- APN 1 - DHCP IP Alloc
    - UDP Downlink 250kbps
    - 100 Subs
- APN 2 - Static IP Alloc
    - UDP Downlink 250kbps
    - 100 Subs
- Attach Rate - 3 UEs/sec

NOTE: If more than 2 APNs per subscriber are connected concurrently, some data flows are not programmed on the data path properly. This is currently under investigation and tracked under github issue #7867

NAT



- eNBs 12
- HTTP Downlink 750kbps per UE
    - 600 UEs
- Attach Rate - 5 UEs/sec
- Peak Tput - +400mbps

Router Mode

Multi-APN with 3APNs; but one APN per subscriber.

APN1 - Static, 100 subs, 250kbps HTTP DL
APN2 - DHCP, 100 subs, 250kbps HTTP DL
APN3 - Static + Router Mode (untagged traffic), 100 subs, 250kbps HTTP DL

```
root@phy-u7:/home/magma# mobility_cli.py get_def_gw
GW IP 10.22.70.1 MAC: 00:00:0c:9f:f1:0e vlan NO_VLAN
GW IP 10.22.168.1 MAC: 00:0c:29:d2:b5:b4 vlan 168
GW IP 10.22.128.4 MAC: 00:50:56:87:c8:41 vlan 128
```

NOTE: Routed UE subnet follows the AGW default route without any vlan tags, however, static and dhcp subnets will use tags 168 and 128 respectively.

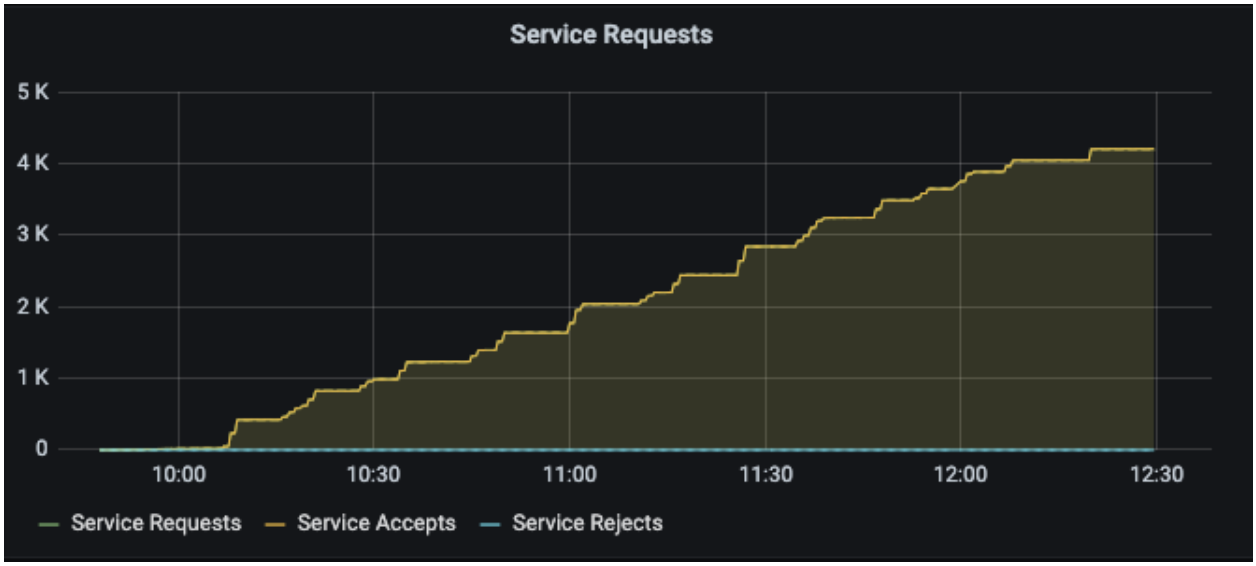
Peak traffic - 75mbps (25 + 25 + 25) across all three APNs.

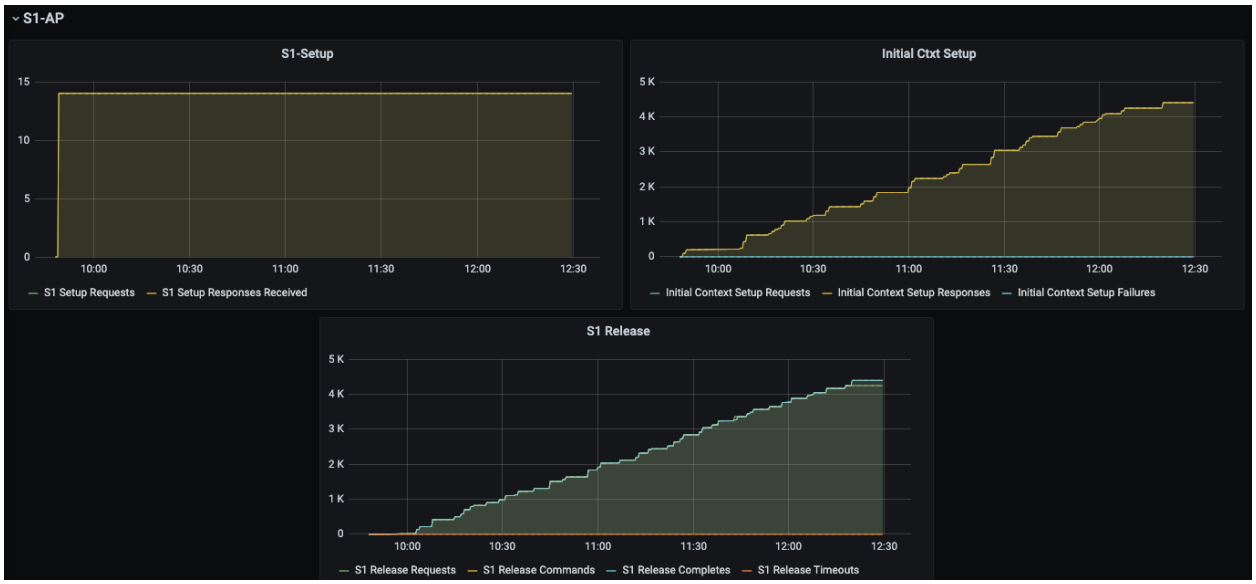
Inbound Roaming

Ubuntu - 200 Subs, 500kbps, 5UEs/sec, Active/Idle transition tested, 12eNBs, 40% roaming

Note: GTP-U Echo Response ***is not currently*** supported in this release. Inactivity timers on the PGW's S8 configuration should be set relatively high (~10 minutes) so as to not drop the bearer. This behavior was noted during the active/idle testing; in the lab environment, inactivity timer was defined as 3 missed Echo requests (~150s). During this time if there was no traffic from/to the UE or an Echo response from the Access Gateway, the PGW dropped the S8 bearer and blackholed traffic for the subscriber. If this behavior is observed, the device must re-attach to regain access to the internet.


DL Tput
125
100
75
50
25
0
10:00
10:30
11:00
11:30
12:00
12:30
Server Sent (Mbps)
Client Received (Mbps)
UL Tput
4.0
3.0
2.0
1.0
0
10:00
10:30
11:00
11:30
12:00
12:30
Server Received (Mbps)
Client Sent (Mbps)
EMM
Attach
250
200
150
100
50
0
10:00
10:30
11:00
11:30
12:00
12:30
Attach Requests
Attach Accepts
Attach Rejects
Attach Timeouts
Auth
250
200
150
100
50
0
10:00
10:30
11:00
11:30
12:00
12:30
Auth Requests
Auth Responses
Auth Rejects
Detach (From Spirent's Perspective)
150
100
50
0
10:00
10:30
11:00
11:30
12:00
12:30
Detach Requests Sent
Detach Requests Received
Detach Responses Sent
Detach Responses Received

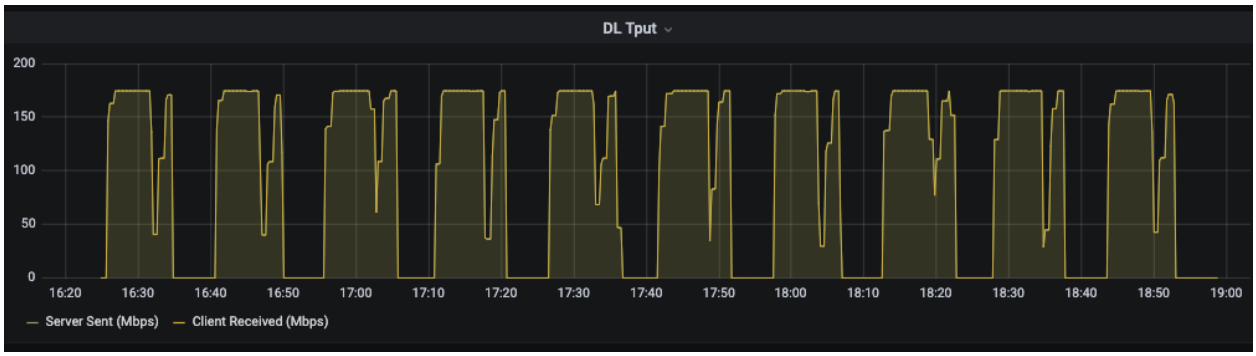




QOS Profiles / Policies

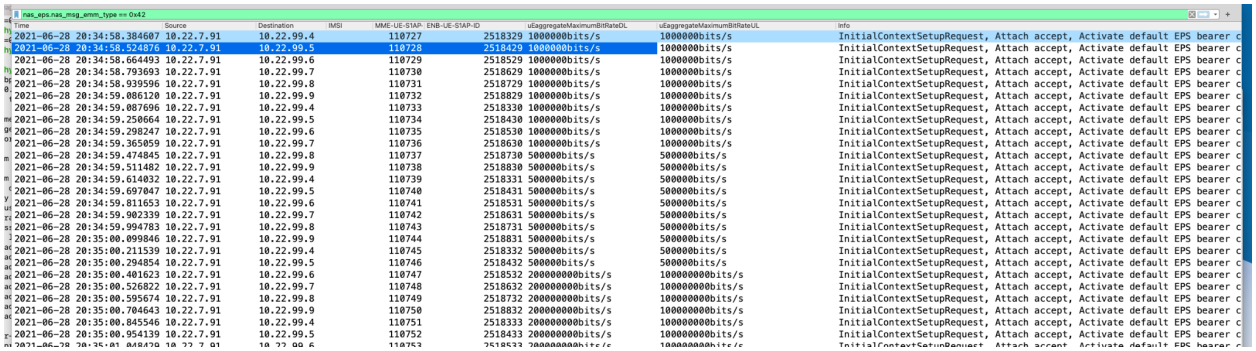
APN AMBR

- 200 Subs
- HTTP 10M Per UE
- APN AMBR - 1mbps; Aggregate TCP Rate peaked at 175Mbps that is 875Kbps per UE.



UE AMBR

Captured pcaps for 3 different data plans bronze (500Kbps DL/UL), silver (1Mbps DL/UL) and default (200/100 Mbps DL/UL).



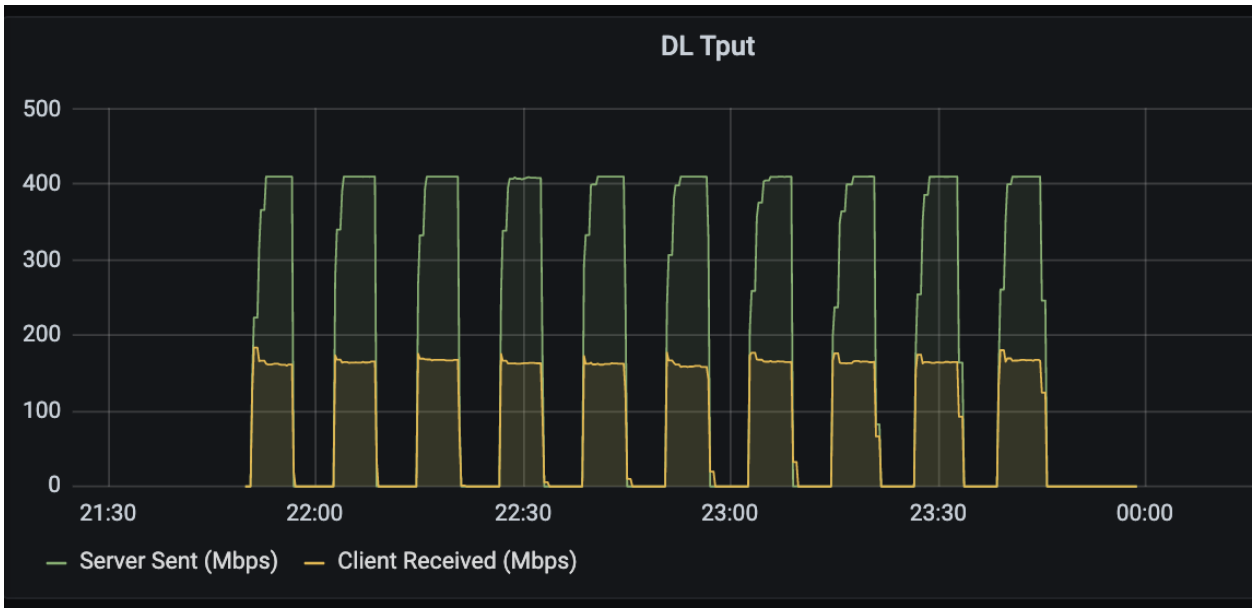
FLOW RESTRICTION

Flow QoS Policies:
50 UEs with TCP traffic rate limited to 1Mbps UL/DL
50 UEs with TCP traffic rate limited to 0.5Mbps UL/DL
50 UEs with UDP traffic rate limited to 1Mbps UL/DL
50 UEs with UDP traffic rate limited to 0.5Mbps UL/DL

APN-AMBR: 2Mbps UL/DL

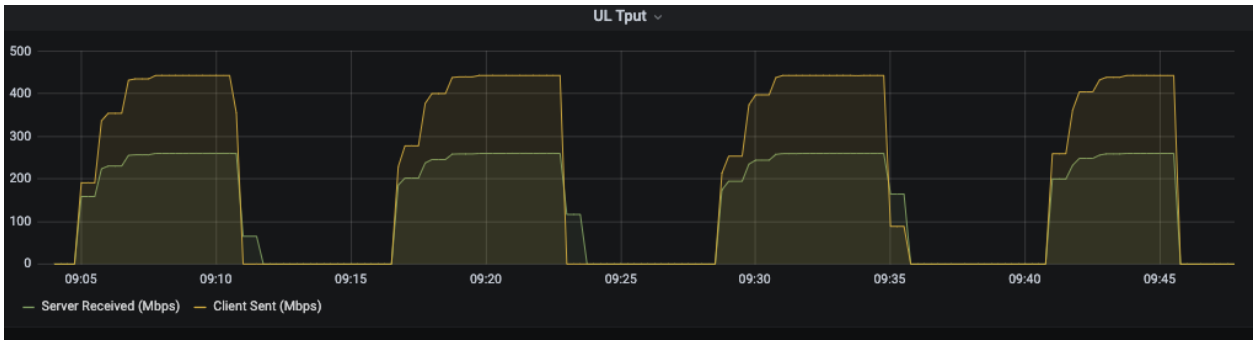
- Test executed:
    - Total Subs - 200
    - Traffic Profile - UDP DL 2M per UE

Expected outcome: DL traffic < 275Mbps
Actual outcome: DL traffic throttled at about 165Mbps



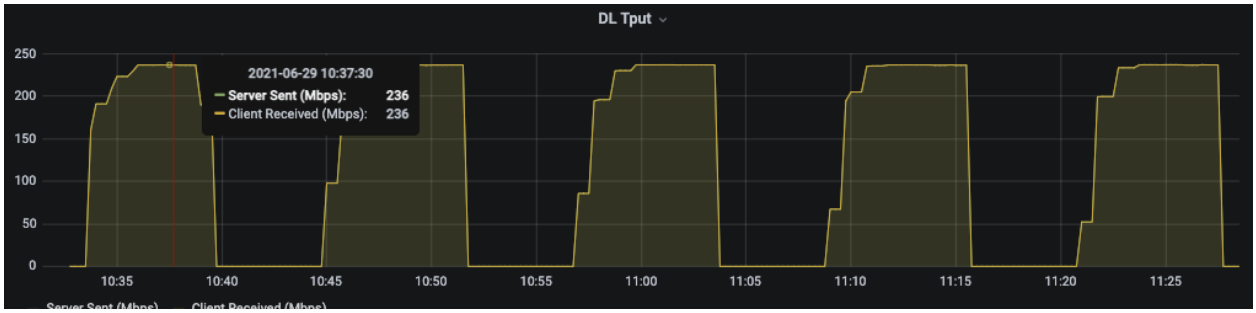
- Test executed:
    - Total Subs - 200
    - Traffic Profile - UDP UL 2M per UE

Expected outcome: UL traffic < 275Mbps
Actual outcome: UL traffic throttled at about 259Mbps



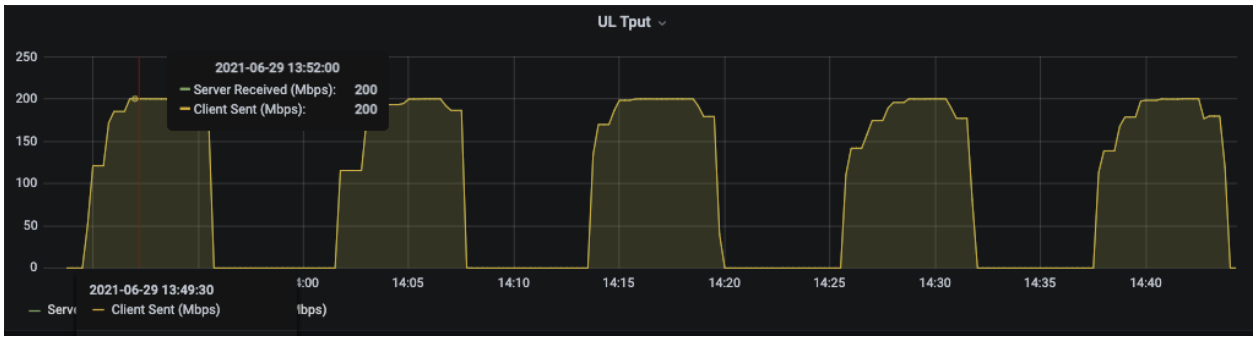
- Test executed:
    - Total Subs - 200
    - Traffic Profile - HTTP DL 5M per UE

Expected outcome: DL traffic < 275Mbps
Actual outcome: DL traffic at 236Mbps (for TCP traffic 85% efficiency looks inline with other experiments)



- Test executed:
    - Total Subs - 200
    - Traffic Profile - HTTP UL 2M per UE

Expected outcome: UL traffic < 275Mbps
Actual outcome: UL traffic throttled at about 200Mbps

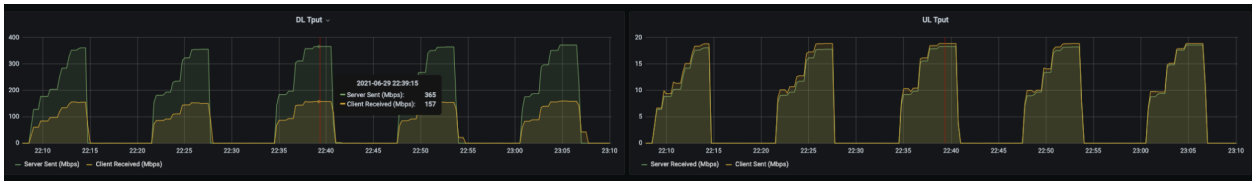


Flow QoS Policies

Each Subscriber has one HTTP and one UDP traffic rule each limited with 500K UL/DL

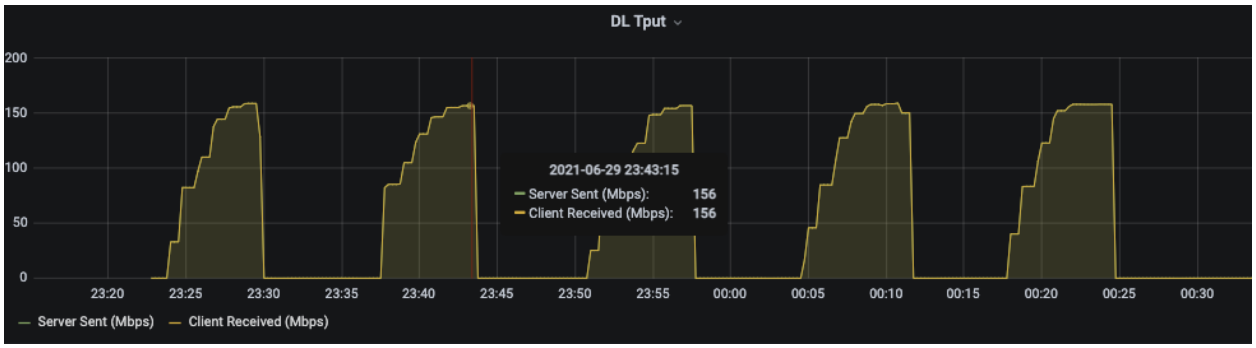
- Test executed:
    - Total Subs - 400
    - Attach Rate - 5UEs/sec
    - Traffic Profile - UDP DL 1M per UE

Expected outcome: DL traffic < 200Mbps
Actual outcome: DL traffic throttled at about 157Mbps [Note that server sending rate does not reach to 400Mbps expected value but at 365Mbps, also note significant ~19Mbps UL traffic)



- Test executed:
    - Total Subs - 400
    - Attach Rate - 5UEs/sec
    - Traffic Profile - HTTP DL 1M per UE

Expected outcome: DL traffic < 200Mbps
Actual outcome: DL traffic throttled at about 156Mbps



Header Enrichment

- Test executed:
    - Total Subs - 300
    - Attach Rate - 3UEs/sec
    - Traffic Profile - UDP DL 1M per UE
    - One HE rule.

HTTP Header enriched for all UE after a couple of HTTP request attempts. It was validated using packet capture on the SGi interface and tshak was used to parse HTTP headers.

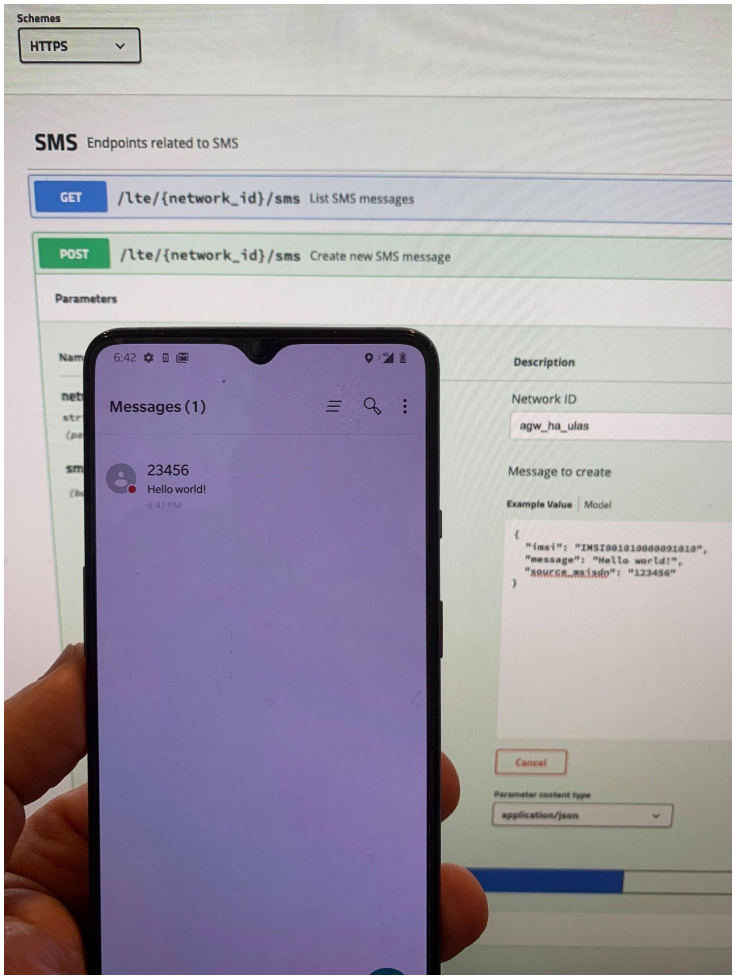
Show Tech

```
# On your GW host, run the following command as user
root:
# If you have git repo checked out already
$ cd ${MAGMA_ROOT}/show-tech
$ ansible-playbook show-tech.yml


# In case you want to download and process latest
version of this playbook from Magma's master:
$ ansible-pull -U https://github.com/magma/magma.git
show-tech/show-tech.yml -d /tmp/show-tech --purge
```

The captured output is dumped in **/tmp/magma_reports/report.magma.<date>.tgz**

SMS delivery



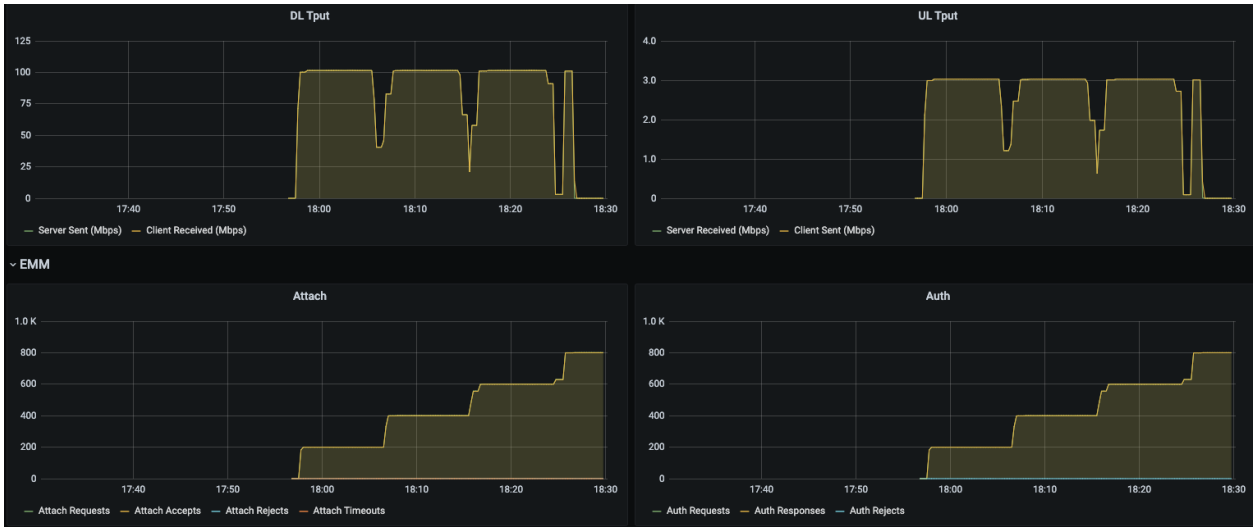
APN correction

Source	Delta Time	Destination	Protocol	IMSI	Info	APN
10.22.99.4	16:45:32.044410	10.22.7.96	S1AP		S1SetupRequest	
10.22.7.96	16:45:32.044622	10.22.99.4	SCTP		SACK	
10.22.7.96	16:45:32.050471	10.22.99.4	S1AP		S1SetupResponse	
10.22.99.4	16:45:32.150347	10.22.7.96	SCTP		SACK	
10.22.99.4	16:45:34.044525	10.22.7.96	S1AP/NAS-E…	001011234560000	InitialUEMessage, Attach request, PDN connectivity request	
10.22.7.96	16:45:34.076380	10.22.99.4	S1AP/NAS-E…		DownlinkNASTransport, Authentication request	
10.22.99.4	16:45:34.076499	10.22.7.96	S1AP/NAS-E…		UplinkNASTransport, Authentication response	
10.22.7.96	16:45:34.084900	10.22.99.4	S1AP/NAS-E…		DownlinkNASTransport, Security mode command	
10.22.99.4	16:45:34.085326	10.22.7.96	S1AP/NAS-E…		UplinkNASTransport, Security mode complete	
10.22.7.96	16:45:34.091369	10.22.99.4	S1AP/NAS-E…		DownlinkNASTransport, ESM information request	
10.22.99.4	16:45:34.091721	10.22.7.96	S1AP/NAS-E…		UplinkNASTransport, ESM information response	internet
10.22.7.96	16:45:34.170113	10.22.99.4	S1AP/NAS-E…		InitialContextSetupRequest, Attach accept, Activate default EPS bearer context request	oai.ipv4
10.22.99.4	16:45:34.170368	10.22.7.96	S1AP		InitialContextSetupResponse	
10.22.99.4	16:45:34.170370	10.22.7.96	S1AP/NAS-E…		UplinkNASTransport, Attach complete, Activate default EPS bearer context accept	

NOTE: Empty APN requests cannot be modified, however, an APN requested by the UE can be updated using the APN correction feature.
NOTE: APN requested by UE was “internet” but was overridden in the Attach Accept message to “oai.ipv4”

Longer Test -

./apn_override.py -s phy-u6 -S 200 -e 12 -a 5 -w 500 -t 45 (RID 537 - phy-u6)



Gx & Gy interface support

Test	TeraVM script	status
Basic Init/Terminate 1 UE, Dynamic rules only	gx_gy_combined_01	OK
Basic Init/Terminate 1 UE, static rules only	gx_gy_combined_02	OK
Basic Init/Terminate 1 UE, Static/Dynamic rule mixture	gx_gy_combined_03	OK
Basic Init/Terminate 32 UEs, Static/Dynamic rule mixture	gx_gy_combined_04	OK
Quota Exhaustion, 1 default rule - 1 UE	gx_gy_combined_05	OK
Quota Exhaustion, 1 default rule - 1 UE (bigger grant)	gx_gy_combined_05_XL	OK

Quota Exhaustion, 2 differentiated rules - 1 UE	gx_gy_combined_06	OK
Rule Level Usage Monitoring updates to PCRF - 1 UE	gx_gy_combined_08	OK
Rule Level Usage Monitoring updates to PCRF - 1 UE (bigger grant)	gx_gy_combined_08_XL	OK
Rule Level and Session Level Usage Monitoring updates to PCRF - 1 UE	gx_gy_combined_09	OK
OCS charging and PCRF usage monitoring mix - 1 UE	gx_gy_combined_10	OK
Rule installation - 1 UE	gx_gy_combined_11	OK
Remove blocking rule with RAR https://our.intern.facebook.com/intern/tasks/?t=39921735	gx_gy_combined_12	OK
Rule installation, removal, and modification mix - 1 UE	gx_gy_combined_14	OK
OCS Failure Scenario - No quota available for the user (transient failure code = 4012 or GSU =0 with result code = SUCCESS)	gx_gy_combined_24	OK
PCRF Failure Scenario - user unknown	gx_gy_combined_26	OK
PCRF Failure Scenario - no rules returned	gx_gy_combined_27	OK
PCRF Failure Scenario - reauth for unknown subscriber	gx_gy_combined_28	OK

CDR export

PLMN restriction

Tested with OnePlus and Baicells eNB. Added MCC=001 and MNC=01 for PLMN restriction list, observed **reject** with the proper cause. Removed the PLMN from the list, forced reattach with airplane mode off/on and observed a successful **attach** with data connectivity.

IMEI restriction

Tested with OnePlus and Baicells eNB. Validated attach rejection with matching TAC (SNR wildcarded) and TAC+SNR as well as attach accept with no matching TAC and TAC+SNR.

IPFix records export

Configuration file changes:

magmad.yml

```
<truncated>

# List of services for magmad to control
magma_services:
  - control_proxy
  - subscriberdb
  - mobilityd
  - directoryd
  - enodebd
  - sessiond
  - mme
  - pipelined
  - envoy_controller
  - redis
  - dnsd
  - policydb
  - state
  - eventd
  - smsd
  - ctraced
  - health
  - connectiond

<truncated>
```

connection.yml (new file)

```
---
#
## Copyright 2020 The Magma Authors.
#
## This source code is licensed under the BSD-style license found in the
## LICENSE file in the root directory of this source tree.
#
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.

log_level: INFO

## Interface for internal packet sending
interface_name: ipfix0


# IMPORTANT when modifying also modify the corresponding pipelined.yml
entry
zone: 897

## Used for generated internal packets
pkt_dst_mac: "33:aa:99:33:aa:00"
pkt_src_mac: "55:11:44:ee:00:00"
```

pipelined.yml

```
<truncated>

conntrackd:
  enabled: true


ipfix:
  enabled: true
  probability: 65
  collector_set_id: 2
  collector_ip: '10.22.3.240' #IP of the collector
```

```
  collector_port: 65010
  cache_timeout: 60
  obs_domain_id: 2
  obs_point_id: 2

<truncated>
```

Added nprobe onto the destination host (ran it in listener mode `sudo nprobe -i ens160 -3 65010 V 10  -P /home/magma/ipfix/ )`

Test Run:

- 100 UEs, 1 HTTP GET request each
- Observed 223 packets (100 reqs * 2 (start/end of flow) + ~33 template files)
- Records generated as expected.

Metrics

Alerts

Alerts	Status
Cpu Percent Alert	OK
Sctpd Crashlooping Alert	OK
S6A Auth Failure	OK
Gateway Checkin Failure	OK
UE attach Failure	OK
High Memory Usage Alert	OK
Service Crashlooping Alert	OK
Unexpected Service Restart Alert	OK
High Disk Usage Alert	OK
Service Restart Alert	OK
S1 Setup Failure	OK
Dip in Connected UEs	OK
Bootstrap Exception Alert	OK
Certificate Expiring Soon	OK
High duplicate attach requests	OK

Validation_alert	OK

NMS Validation

NMS	Action	Status
NMS	Create an organization	OK
NMS - Administrative tools	Create a user	OK
NMS - Network	Create a network	OK
NMS - Equipment - Gateway	Create a gateway	OK
NMS - Subscriber	Create a subscriber	OK
NMS - Subscriber	Edit a Subscriber	OK
NMS - Traffic - Policies	Create a profile	OK
NMS - Traffic - Policies	Create a policy	OK
NMS - Traffic - APN	Create an APN	OK
NMS - Traffic - Policies	Create an Rating group	OK
NMS - Network	Modify a network	OK
NMS - Network	Enable NAT	OK
NMS - Network	Enable DHCP	OK
NMS - Network	Enable Static IP	OK
NMS - Equipment - Gateway	Modify a gateway	OK
NMS - Equipment - eNodeB	Add an unmanaged eNB	OK
NMS - Equipment - eNodeB	Add an managed eNB	OK
NMS - Equipment - Gateway	Edit Aggregation	OK
NMS - Equipment - Gateway	Edit EPC	OK
NMS - Equipment - Gateway	Edit RAN	OK
NMS - Equipment - Gateway	Edit Header Enrichment	OK
NMS - Call Tracing	Create call trace	OK
NMS - Call Tracing	Download call trace	OK
NMS - Metrics	Check Metrics	OK
NMS - Alerts	Configure Alerts	OK
NMS - Alerts	View Alerts	OK
NMS - Alerts - receivers	Configure Alert receivers	OK

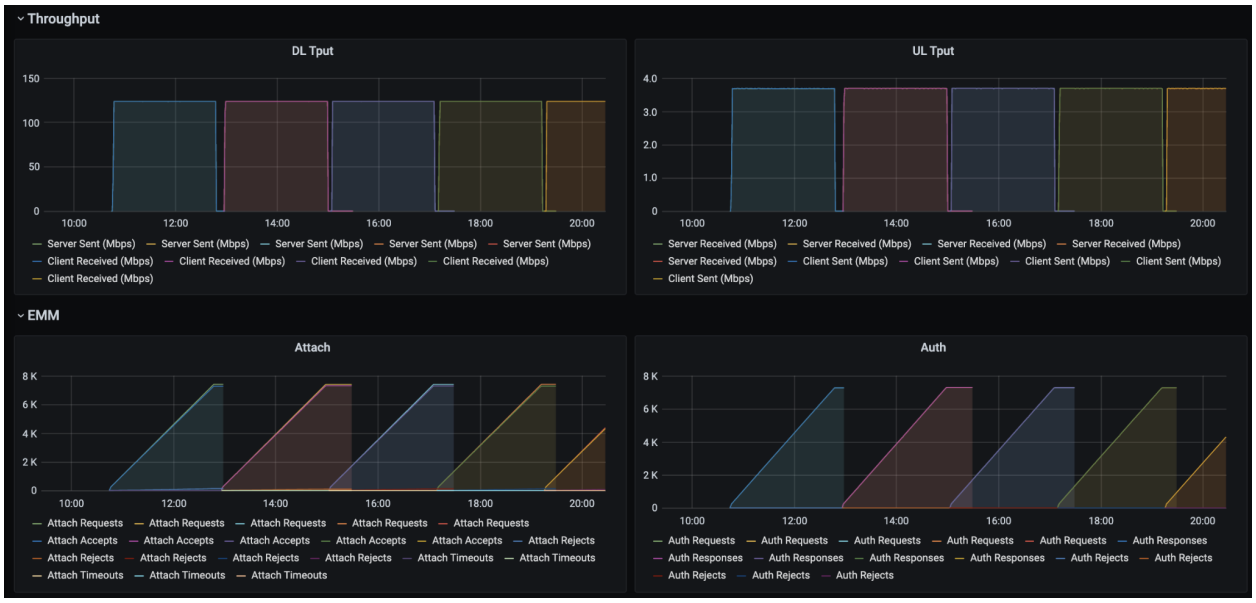
Stability Testing (12 hrs)

NAT

Ubuntu Phy

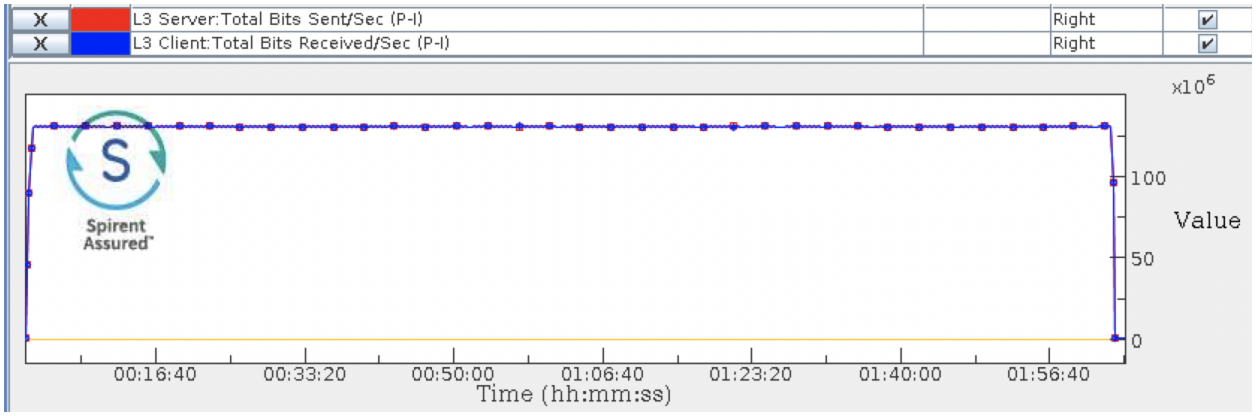
Test:

- Part1:
  - Get Magma on normal running load with 240 UEs and send traffic 500Kbps continuously
- Part2:
  - Continuously attach new subs at 1UE/sec and detach UE at 1 UE/sec for next 2 hours.
  - Verify ping send/received for each UE attached
- Run both part of test over loop for 12 hours



Constant System peak traffic - 120Mbps = 240 x 500 kbps

Each run details looks as below:
Constant DL traffic:



Constant UE attach/detach: