

5G Testbed for MEC Implementation

Ans Fida, Haotian Deng, Mohammad Haseeb

Abstract

This project presents an implementation-based study of the concept of Mobile Edge Computing (MEC). We took a deploy-implement-experiment approach in this project. We first deploy a 4G LTE network testbed using the open-source software OpenAirInterface. Next, we implement a basic MEC controller that is intelligent enough to route UE traffic destined for MEC to MEC and all other traffic to the Internet via EPC. Finally, we compare and contrast the difference in latencies added by the OAI EPC core code. Our results show that accessing the MEC directly is 7-10 milliseconds faster on average as compared to accessing it via the EPC. This has been tested on our testbed and on a local area network. This project has been an effort towards paving the way for our eventual goal of implementing a *carrier independent MEC*.

1. Introduction

There is exciting research going on in both the telecommunication industry and the academia regarding the new generation mobile network: 5G. Many new technologies have been proposed, some of which will help achieve the 5G network we have defined while others will compliment user experience and lower the load on network providers. NFV, SDN, cRAN, mmWave etc. are all designed while keeping the 5G KPIs in mind. Along with these new solutions, the concept of Mobile Edge Computing (MEC) is another promising concept that will cater to the emerging delay-sensitive services, such as real-time virtual reality, safe autonomous driving, remote healthcare monitoring, IoT, etc.

MEC offers cloud-computing capabilities at the edge of the network to application developers and content providers. It allows the availability of the cloud servers inside or adjacent to the base stations. This ensures ultra-low latency

and high bandwidth as well as real-time access to radio network information that can be leveraged by different applications to increase performance [1]. By providing low latency requirements to users and only selectively sending data over the core network to the remote cloud, MEC provides an attractive platform for users, application developers, and network operators alike. The European Telecoms Standards Institute (ETSI) has the responsibility of coming up with standards for MEC and aiding its timely development.

The purpose of our work has been to verify the feasibility of MEC and an MEC controller, quantify its performance aspects, and find control-plane related information and functions that might be needed to build the next generation MEC i.e. a carrier independent MEC. However, before we could begin working on MEC, we had to have some infrastructure or a testbed on which to build and test our hypothesis. For this, we leveraged OpenAirInterface (OAI), which is an open-source software implementation of the 4G LTE network stack and entities. We explain OAI and our testbed setup in detail in section 2. We specifically looked into how a mobile network operator might implement the routing of user data at the eNodeB and what are the limitations of the different approaches. In addition to this, we quantified the performance gain of MEC over OAI implementation. We present our solution for packets routing at MEC controller and performance assessment in sections 3 and 4, respectively. We wrap our paper by mentioning some of the future directions for this work and finally, providing some concluding remarks on our work.

2. Testbed Setup

The concept of building a mobile network testbed for experimental purposes has gained momentum in recent years. Industry giants like Qualcomm and Deutsche Telekom have their own

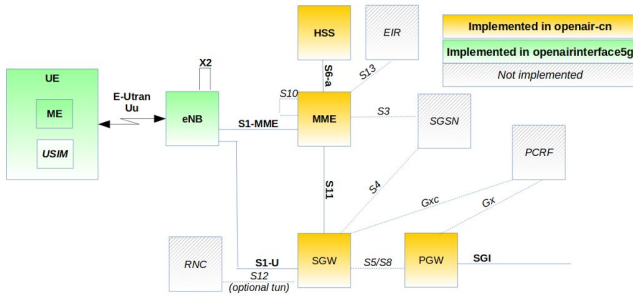


Figure 1: OAI Entities

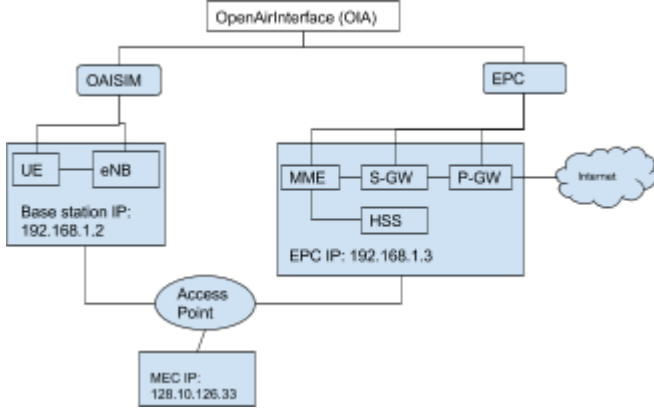


Figure 2: Testbed Architecture

testbed implementations [2]. There are also some publicly available testbeds such as 5G Berlin [3]. They offer many services related to 5G including 5G Playground, Open5GCore and OpenSDN Core. In academia, NYU Wireless Lab announced last year that they will be building “an advanced programmable platform to support the development of millimeter wave (mmWave) wireless communication” [4]. Prior to us, Professor Guan-Hua Tu at Michigan State University (MSU) had setup an OAI testbed and we got some initial help from their work.

OAI is an open-source software-based implementation of the 4G LTE network. Figure 1 shows the different 4G LTE network entities implemented in the OpenAirInterface software [5]. OAI can be used to build and customize an LTE base station (OAI eNB), a user equipment (OAI UE) and a core network (OAI EPC) on a machine running Linux. OAI provides separate modules for each of the above entities but also provides a UE simulator that emulates an actual UE. In this case, the eNB and the UE code are bundled together into the module called OAISIM. Although we originally wanted to use an actual UE along with a USRP, we could not immediately get hands on a programmable SIM card that is

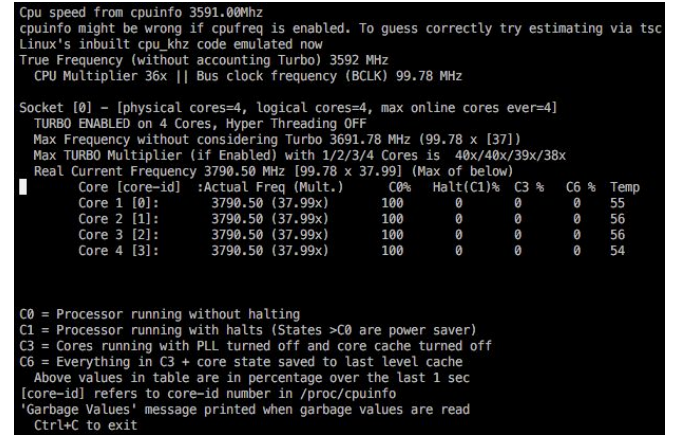


Figure 3: eNB Machine CPU configuration

required for setting up the UE and hence, decided to go with the UE simulator provided by OAI. The are multiple tutorials [6][7][8] from the OpenAirInterface community which help us a lot during our deployment.

We now talk about the underlying hardware of our testbed and then give an in-depth explanation of the software side of the testbed. Note that we have three machines in our testbed setup: (1) eNB and UE machine with IP 192.168.1.2, (2) EPC machine with IP 192.168.1.3 containing the MME, HSS, S-GW and P-GW modules and (3) MEC machine with IP 128.10.126.33. We use the MSSN2 server to serve as our MEC for now. Figure 2 shows a high level overview of our testbed architecture.

2.1. Hardware Configurations

OAI provides numerous options when it comes to running the eNB and the UE. Among others, some of the options the user has are selecting various radio settings for the eNB, whether to use a physical UE with a USRP or using a built in simulator for the UE, etc. Since the eNB has to emulate many LTE network operations from the physical layer up to the application layer, the eNB module of OAI is a compute-intensive piece of software that requires fast hardware. For this reason, we had to turn off a number of CPU related checks on the eNB machine which would allow all cores on the machine to run at full capacity. We disabled all power management features (HyperThreading, CPU frequency control, C-States, P-States). Moreover, we disabled CPU Frequency scaling to enable all the

- Protocol stacks overview and S1U/S5 encapsulation

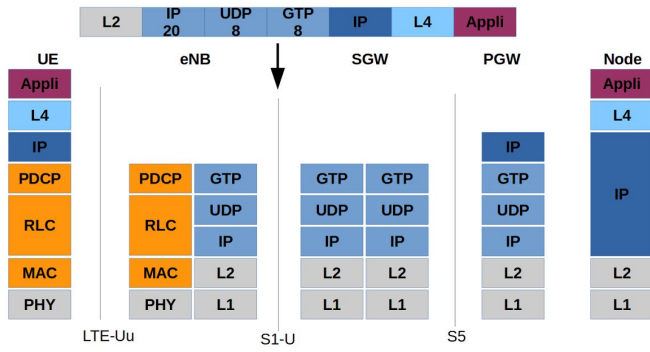


Figure 4: OAI Protocol stack overview

cores to function at their highest frequencies. We verified that all cores were operating at full capacity using an Ubuntu package called *i7z*. The output is shown in Figure 3. It shows that all four cores of our eNB machine are operating at their maximum frequencies and they are all running without halting. Additionally, it verifies that Hyperthreading has been turned off.

2.2. Software Configurations

OAI is primarily designed to be run on Ubuntu. Although the tutorial we were following mentioned to use Ubuntu 14.04 for eNB and Ubuntu 16.04 for EPC, we found that having different operating system versions on both the entities caused errors while exchanging initial SCTP set up messages between the two machines. We then switched to using the same Ubuntu versions (16.04) on both the machines. This solved the SCTP mismatch problem. As mentioned before, the eNB code simulates many RAN functions and therefore, requires ultra-low latencies. Consequently, the eNB code also requires the use of a low-latency kernel instead of the general kernel shipped in an Ubuntu distribution. The same holds true for the EPC code, however, that required a special kernel version made for the OAI EPC.

Having setup the hardware and the operating system at each host, the next step was to configure the actual OAI entities, namely the MME, HSS, Serving and Packet Gateway (SPGW) and the OAI-SIM (eNB and UE). OAI provides predefined configuration files for each of these entities and it is the developer's responsibility to correctly configure the IP

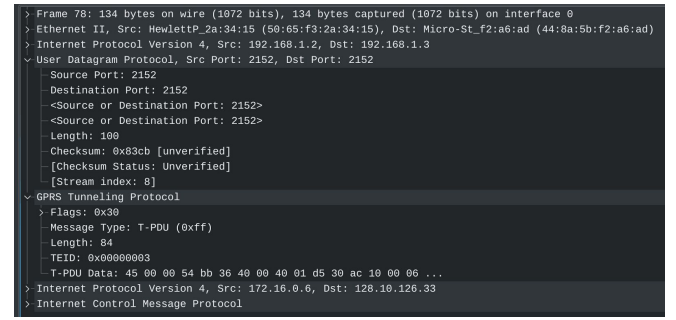


Figure 5: An example of ICMP packets transferred through S1-U link.

addresses, port numbers, host names, and database entries for each entity's configuration file.

This completed our software configuration for the OAI and after this, the following interfaces were initialized between their respective entities: (1) S6-a between HSS and MME, (2) S11 between MME and SGW, (3) S5/S8 between SPGW, (4) S1-MME between eNB and MME and (5) S1-U between eNB and SGW. Moreover, the UE was successfully able to PING any host on the Internet using the virtual network interface provided by the OAI-SIM configuration. A Wireshark packet capture on the EPC machine verifies that the packets from the UE to any Internet host are indeed being routed through the EPC machine. Appendix A contains the screenshots of the whole flow from launching the HSS to the UE accessing the Internet through a PING command on the virtual network interface (having IP address 176.16.0.2).

3. MEC Controller

In order to support MEC server service along with the current mobile system so that computing power and media content required by latency sensitive applications can be brought from cloud to edge, eNodeB is expected to use an extra link to deliver specific user packets to MEC servers deployed at network edge directly instead of routing packets going through the core network and reaching the remote server finally. MEC controller plays the role of filtering out selected user packets and forwarding them to the MEC server here. We explain how we implement our simplified MEC controller and the lesson we learned in this section.

As shown in the Figure 4 about the protocol stack overview, in the implementation of eNodeB side in the OpenAirInterface project, all user plane packets coming from user equipment will be decapsulated from physical layer till link layer and stopped before the IP layer. The eNodeB will then treat the user IP packets as an application layer packet and encapsulate it with a GTP (GPRS Tunnelling Protocol) header, a UDP header (to indicate the SGW port on EPC machine) and another IP header (to indicate the EPC machine IP address). The final new IP packets generated by eNodeB implementation are sent out to the core network through the virtual network interface oip1 which is created by OAI for internal transmission. The core network code of OAI will receive and handle this IP packets and extract the inside user IP packets and finally deliver it out to the Internet. Figure 5 shows an example of an ICMP (generated by ping command) packet originally from UE and go through the S1-U link to the S-GW. The inner user IP packets contains the ICMP packet and has the 172.16.0.6 (the virtual private IP address assigned to the UE simulator) as source IP address and 128.10.126.33 (MSSN2 as the MEC server) as the destination IP address. The outer IP packet header encapsulated by the eNodeB uses 192.168.1.2 (local IP address of eNodeB machine) as the source address and 192.168.1.3 (local IP address of EPC machine) as the destination address. The port number 2152 in the UDP header is designated for S1-U link between eNodeB and S-GW. The basic responsibility of MEC controller is filtering out such user IP packets targeted to the MEC server and reroute it to the MEC server directly by modifying the outer IP packet destination address to the MEC server address.

3.1. IPTables

Our first approach is utilizing iptables command on the Linux machine which host eNodeB to redirect those specific user traffic packets. The linux tool iptables command is an administration tool for IPv4/IPv6 packet filtering and NAT (Network Address Translation). The original expectation is using the u32 module in iptables to

match the packets with specific pattern and apply routing rules for those packets. However, it turns out that all redirect rules specified in iptables applied to packets which go through oip1 interface will be overwritten by the eNodeB implementation. An interesting finding is that we are capable to drop those packets locally but any redirection fails to take effect. Such observation makes us believe the eNodeB side implementation in OAI overwrite over any iptables rules for packets go through the oip1 interface.

Our second approach is to bring another machine as the physical MEC controller entity to the table. Our target is using this physical machine to separate packets coming from eNodeB and redirecting them to either MEC server or the EPC machine. By modifying the configuration of the OAISIM, we are able to specify the IP address used for S1-MME interface to the MEC controller machine and the registrations procedure between eNodeB and the core network succeed after we use iptables to route all incoming packets from OAISIM machine to EPC server at the MEC controller. However all the following user traffic still goes to the EPC machine directly. The reason is the eNodeB side get the IP address used for S1-U interface to S-GW through the inner communication with MME. Such behaviour cannot be manipulated simply through changing configurations at eNodeB side. Based on this understanding we also tried to modify the IP address for S11 interface between MME and S-GW to the MEC controller machine. Unfortunately such changes break the implementation of the openEPC as the EPC server are no longer reachable to the MEC controller machine.

3.2. Routing Code

As the previous approach of using iptables command did not fulfill our expectation, we implemented our own filtering and routing functionality at the GTP layer at the eNodeB side implementation. The basic idea is straightforward: in the GTP implementation, we locate the range of bytes from the GTP packet payload which representing the destination address of the user IP

```

> Frame 3: 134 bytes on wire (1072 bits), 134 bytes captured (1072 bits) on interface 0
> Ethernet II, Src: HewlettP_2a:34:15 (58:65:f3:2a:34:15), Dst: Netgear_ce:ab:76 (08:02:8e:ce:ab:76)
> Internet Protocol Version 4, Src: 192.168.1.2, Dst: 128.10.126.33
> User Datagram Protocol, Src Port: 2152, Dst Port: 2152
> GPRS Tunneling Protocol
> Internet Protocol Version 4, Src: 172.16.0.4, Dst: 128.10.126.33
> Internet Control Message Protocol

```

Figure 6: An example of ICMP packet received at MEC server directly from eNodeB.

packets by calculating the proper offset. This offset is fixed as the header length for both GTP headers (8 bytes) and the user IP packets header (20 bytes) implemented in the OAISIM is fixed. So this payload range for the user IP packet destination address is from the 25th byte to the 28th byte. After filtering out user IP packets with destination address to the MEC server, we send it over UDP message by assigning the destination address to MEC server instead of S-GW.

Our patch code are applied to the `gtpv1u_send_udp_msg()` function in the `openairinterface5g/openair3/GTPV1-U` model. After applying such modification, we managed to redirect all user packets with MEC server as destination to MEC server directly without going through the EPC first.

Figure 6 shows an example of ICMP packet received at the MEC server which is sent out directly from eNodeB. The outer IP packet has the address of MEC server as destination address instead of the S-GW address like other normal user traffic.

4. Preliminary Assessment and Validation

We try to conduct the preliminary study about the latency improvement for routing traffic to edge server without going through the core network based on the the testbed upon OpenAirInterface implementation. The results we presented here could only show the extra latency caused by the openEPC implementation and the value could be even larger in reality.

Following are the detail of the preliminary study procedure. We keep sending ping requests from the user device to the MEC server for 60 seconds and record both timestamps of ICMP request and response packets using Wireshark to calculate the RTT. We compare the latency

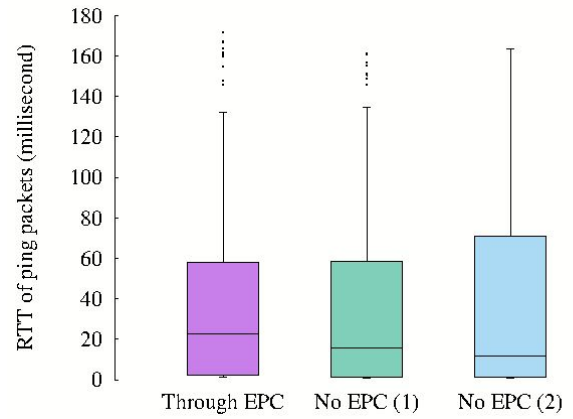


Figure 7: Comparison of latency performance between three cases using RTT of ping request and response messages.

performance between traffic which go to MEC server directly with traffic which go through the core network before reaching the MEC server (case “*Through EPC*”) by enable/disable our MEC controller routing function.

There is an issue need to be mentioned here, the IP packets which contains the ICMP message received at MEC server directly from eNodeB also contains extra headers including one GTP header, one UDP header and another IP header unlike those IP packets received from core network after header decapsulation. So the ping message from user equipment without going through core network cannot get proper response, instead a “destination unreachable” response is received from the MEC server (case “*No EPC 1*”). The source IP address of such response messages actually is the address of the MEC server, so it still can be used to calculate the RTT although we cannot quantify the processing difference between such response with a proper ping reply. In addition to this, we also implement a simplified server at the the MEC machine with Python socket programming. Once a IP packet contains GTP, UDP, IP and ping request inside is received, the server will send the same IP packets back to the user equipment immediately as a response (case “*No EPC 2*”). The wireshark running at UE side can also capture such response and calculate the RTT.

We compare the latency of these three cases. Figure 7 shows the performance gain of the two “No EPC” cases compared with the “Through

EPC” case, both “No EPC” cases get smaller median RTT values with 7 and 10 milliseconds respectively while the “No EPC 2” case has 10 millisecond larger for the upper quartile.

This preliminary study validates that routing traffic to the edge server directly without going through the core network could help to improve the latency. Considering in our testbed, both eNodeB and EPC are deployed under the same local area network, the network delay would be ignorable and most of the saving should be caused by the EPC processing time depends on the OpenAirInterface core network implementation.

5. Future Work

Our main task and purpose in this project was to setup the infrastructure (testbed) on which we implement and test our initial MEC controller. However, some problems like UE mobility, security, privacy and billing are still open and need to be solved. Our next step, therefore, is to design and implement a separate SDN controller that would assist and orchestrate the flow of traffic between the UE, eNB and EPC based on control plane information from the EPC and eNB.

Some of the basic requirements we define for this SDN controller are that it must be extremely lightweight, secure and must only implement the crucial control functions and nothing extra. Deciding on what control plane functions to implement is another open question. The reason for having such strict requirements on the controller is so that we can still ensure and provide the ultra-low latency and high availability benefits promised by MEC while still be able to enhance and provide a new carrier independent MEC.

6. Conclusion

In this project, we first setup an open-source 4G LTE network testbed called OpenAirInterface (OAI). We then implemented and proved the feasibility of the concept of an MEC controller. Furthermore, we measured and assessed the latency of accessing the MEC directly and of

accessing the MEC after going through the EPC. Our results show that the former approach gains 7-10 milliseconds improvement of latency on average based on our OpenAirInterface testbed with both eNodeB and EPC deployed in local area network.

Although much work is still to be done, this work would be the stepping stone for our final goal with this project i.e. implementing and studying the possibility of a *carrier independent MEC*.

REFERENCES

- [1] <http://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing>
- [2] <https://www.qualcomm.com/news/releases/2017/02/21/qualcomm-showcases-5g-leadership-its-first-3gpp-based-5g-new-radio>
- [3] <http://www.5g-berlin.org/482f765f1380172e>
- [4] https://campustechnology.com/articles/2016/01/20/nyu-wireless-to-build-test-bed-for-5g.aspx?adm_garea=news
- [5] http://www.openairinterface.org/docs/workshop/3_OAI_Workshop_20170427/training/CN_training_user_plane.pdf
- [6] How to connect OAI eNB (USRP B210) with COTS UE.
<https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/HowToConnectCOTSUEwithOAIeNBNew>
- [7] How to connect with OAI EPC (on different machines).
<https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/T/howtoconnectoaisimwithoaipec>
- [8] OpenAirInterface System Emulation (aka oaisim).
<https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/OpenAirLTEEmulation>

APPENDIX A

• HSS Launch

```
openair@msi-laptop: /home/openair-cn/scripts
openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts...
IMSI: 208920100001120Key: Fe-c8.6b.a6.eb.76.7e.d0.89.05.75.7b.1b.b4.4b.8f.
OPC: c4.24.49.36.3b.ba.d0.2b.66.d1.6b.c9.75.d7.7c.c1.
RjndaeKeySchedule: K FEC86BA6E707ED08905757B1BB448BF
Compute opc:
K: FEC86BA6E707ED08905757B1BB448BF
In: 1006020F0A478BF06899F15C062E42B3
Rinj: D4224B3931FD58DD0B489A9573F93E72
Out: C42449363BA6D02B66D16BC975D77CC1
Query: UPDATE 'users' SET 'OPC' = UNHEX('c42449363ba6d02b66d16bc975d77cc1') WHERE 'users'.'imsi' = '208920100001120'
IMSI 208920100001120 Updated OPC c42449363ba6d02b66d16bc975d77cc1 -> c42449363ba6d02b66d16bc975d77cc1
0 rows affected
Initializing s6a layer
11/27/17, 20:46:28.483485 NOTI libfdproto '1.2.0' initialized.
11/27/17, 20:46:28.483529 NOTI libgnutls '3.4.10' initialized.
11/27/17, 20:46:28.483812 DBG Core state: 0 -> 1
11/27/17, 20:46:28.483821 NOTI libfdcore '1.2.0' initialized.
11/27/17, 20:46:28.484041 DBG Generating fresh Diffie-Hellman parameters of size 1024 (this takes some time)...
11/27/17, 20:46:28.502240 DBG Loading: /usr/local/lib/freeDiameter/acl_wl.fdx
11/27/17, 20:46:28.502187 NOTI Extension ACL_WL initialized with configuration: '/usr/local/etc/oa/freeDiameter/acl.conf'
11/27/17, 20:46:28.502196 DBG Loading: /usr/local/lib/freeDiameter/dict_nas_mtpv6.fdx
11/27/17, 20:46:28.502242 DBG Dictionary Extension 'MIPv6 NAS-to-HAAA Interaction' initialized
11/27/17, 20:46:28.502240 DBG Loading: /usr/local/lib/freeDiameter/dict_s6a.fdx
11/27/17, 20:46:28.502977 NOTI Dictionary Extension 'S6A from 3GPP standard v.10.5' initialized
11/27/17, 20:46:28.502983 NOTI All extensions loaded.
11/27/17, 20:46:28.503008 NOTI freeDiameter configuration:
11/27/17, 20:46:28.503031 NOTI Default trace level ....: 41
11/27/17, 20:46:28.503026 NOTI Configuration file ....: /usr/local/etc/oa/freeDiameter/hss_fd.conf
11/27/17, 20:46:28.503029 NOTI Diameter Identity .....: hss.openair4G.eur (1:17)
11/27/17, 20:46:28.503032 NOTI Diameter Realm .....: openair4G.eur (1:13)
11/27/17, 20:46:28.503036 NOTI Tc Timer .....: 30
11/27/17, 20:46:28.503041 NOTI Tw Timer .....: 30
11/27/17, 20:46:28.503044 NOTI Local port .....: 3868
11/27/17, 20:46:28.503048 NOTI Local secure port .....: 5868
11/27/17, 20:46:28.503052 NOTI Number of SCTP streams ...: 3
11/27/17, 20:46:28.503056 NOTI Number of clients thr ...: 5
11/27/17, 20:46:28.503058 NOTI Number of app threads ...: 4
11/27/17, 20:46:28.503062 NOTI Local endpoints .....: Default (use all available)
11/27/17, 20:46:28.503065 NOTI Local applications .....: (none)
11/27/17, 20:46:28.503071 NOTI Flags: - IP .....: Enabled
11/27/17, 20:46:28.503080 NOTI - IPv6 .....: DISABLED
11/27/17, 20:46:28.503083 NOTI - Relay app ....: DISABLED
11/27/17, 20:46:28.503085 NOTI - TCP .....: Enabled
11/27/17, 20:46:28.503088 NOTI - SCTP .....: DISABLED
11/27/17, 20:46:28.503092 NOTI - Prof. proto ...: TCP
11/27/17, 20:46:28.503094 NOTI - TLS method ...: Separate port
11/27/17, 20:46:28.503098 NOTI TLS: - Certificate ..: /usr/local/etc/oa/freeDiameter/hss.cert.pem
11/27/17, 20:46:28.503100 NOTI - Private key ..: /usr/local/etc/oa/freeDiameter/hss.key.pem
11/27/17, 20:46:28.503104 NOTI - CA (trusts) ...: /usr/local/etc/oa/freeDiameter/hss.ca.cert.pem (1 certs)
11/27/17, 20:46:28.503118 NOTI - CRL .....: (none)
11/27/17, 20:46:28.503121 NOTI - Priority .....: (default: 'NORMAL')
11/27/17, 20:46:28.503124 NOTI - DH bits .....: 1024
11/27/17, 20:46:28.503126 NOTI Origin-State-Id .....: 154833588
11/27/17, 20:46:28.503133 NOTI Loaded extensions: '/usr/local/lib/freeDiameter/acl_wl.fdx'[/usr/local/etc/oa/freeDiameter/acl.conf], loaded
11/27/17, 20:46:28.503136 NOTI Loaded extensions: '/usr/local/lib/freeDiameter/dict_nas_mtpv6.fdx'[(no config file)], loaded
11/27/17, 20:46:28.503139 NOTI Loaded extensions: '/usr/local/lib/freeDiameter/dict_s6a.fdx'[(no config file)], loaded
11/27/17, 20:46:28.503151 DBG Core state: 1 -> 2
11/27/17, 20:46:28.503152 NOTI Local server address(es): 172.16.0.1[---L-] 192.11.1.3[---L-] 192.21.1.3[---L-] 192.168.1.3[---L-]
11/27/17, 20:46:28.503840 DBG Core state: 2 -> 3
Initializing s6a layer: DONE
```

• MME Launch

```
openair@msi-laptop: /home/openair-cn/scripts
openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts... openair@msi-laptop: /home/openair-cn/scripts...
000165 00000:301007 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 Hss.openair4G.eur: Connection established, (---) TCP,#38->127.0.0.1(3868)
000165 00000:301007 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 SENT to 'hss.openair4G.eur': 'Capabilities-Exchange-Request' 0/257 fsk--- src: '(nll)' len:236 [C:264/L:132,C:296/L:121,C:278/L:112,C:257/L:114,C:257/L:114,C:257/L:114,C:267/L:120,C:267/L:12,C:299/L:112,C:269/L:132,C:265/L:112]
000166 00000:301082 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 Sending 236b data on connection [---] TCP,#38->127.0.0.1(3868)
000167 00000:301109 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 'STATE_WAITCNMACK' -> 'STATE_WAITCEA' 'hss.openair4G.eur'
000168 00000:301154 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 Peer timeout reset to 10 seconds
000169 00000:301122 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 'hss.openair4G.eur' in state 'STATE_WAITCEA' waiting for next event.
000170 00000:301729 7F8211293700 DEBUE S6A me/openair-cn/src/s6a/s6a_task.c:0243 Initializing S6a interface: DONE
000171 00000:301750 7F8211293700 DEBUE MME-AP openair-cn/src/oa/mme/oa_mme.c:0149 MME app initialization complete
000172 00000:301873 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 Thread terminated
000173 00000:301901 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 'STATE_WAITCEA' <-- 'FDEVP_CNK_MSG_RECV' (0x7f81700050f0,232) 'hss.openair4G.eur'
000174 00000:301929 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 RCV from 'hss.openair4G.eur': (no model)0/257 f----- src:'hss.openair4G.eur' len:232 [C:268/L:112,C:264/L:125,C:296/L:121,C:278/L:112,C:257/L:114,C:257/L:114,C:257/L:114,C:267/L:120,C:267/L:12,C:299/L:112,C:269/L:132,C:265/L:112]
000175 00000:301947 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 Iterating on rules of AVP: 'Vendor-Specific-Application-Id'
000176 00000:301954 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 Iterating on rules of COMMAND: 'Capabilities-Exchange-Answer',
000177 00000:301975 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 'hss.openair4G.eur' claims support for a subset of vendor 10415 features.
000178 00000:302012 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 Connected to 'hss.openair4G.eur' (TCP,soc#38), remote capabilities:
000179 00000:302019 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 'Capabilities-Exchange-Answer'
000180 00000:302025 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 Version: 0x01
000181 00000:302031 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 Length: 232
000182 00000:302037 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 Flags: 0x00 (---)
000183 00000:302043 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 Command Code: 257
000184 00000:302049 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 ApplicationId: 0
000185 00000:302058 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 Hop-by-Hop Identifier: 0x67C8BFC6
000186 00000:302064 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 End-to-End Identifier: 0xC533AA19
000187 00000:302070 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 [internal data]: src:hss.openair4G.eur(17) rwb:(nll) rt:2 cb:(nll),(nll),(nll) qry:0x7f81d8013010 asso:0 sess:
000188 00000:302076 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Result-Code'(268) l=12 f=M val='DIAMETER_SUCCESS' (2001 (0x7d1))
000189 00000:302082 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Origin-Host'(264) l=25 f=M val='hss.openair4G.eur'
000190 00000:302088 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Origin-Realm'(296) l=21 f=M val='openair4G.eur'
000191 00000:302094 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Origin-State-Id'(278) l=12 f=M val=151182548 (0x5a1cac44)
000192 00000:302100 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Host-IP-Address'(257) l=14 f=M val=192.11.1.3
000193 00000:302105 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Host-IP-Address'(257) l=14 f=M val=192.21.1.3
000194 00000:302111 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Host-IP-Address'(257) l=14 f=M val=192.168.1.3
000195 00000:302117 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Vendor-Id'(266) l=12 f=M val=0 (0x0)
000196 00000:302122 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Product-Name'(269) l=28 f=M val='freeDiameter'
000197 00000:302127 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Firmware-Revision'(267) l=12 f=M val=10200 (0x27d8)
000198 00000:302134 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Vendor-Specific-Application-Id'(260) l=32 f=M val=(grouped)
000199 00000:302140 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Auth-Application-Id'(258) l=12 f=M val=16777251 (0x1000023)
000200 00000:302146 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Vendor-Id'(266) l=12 f=M val=10415 (0x28af)
000201 00000:302154 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 AVP: 'Supported-Vendor-Id'(265) l=12 f=M val=10415 (0x28af)
000202 00000:302160 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 No TLS protection negotiated with peer 'hss.openair4G.eur'.
000203 00000:302185 7F81E97FA700 ERROR S6A me/openair-cn/src/s6a/s6a_task.c:0080 'STATE_WAITCEA' -> 'STATE_OPEN' 'hss.openair4G.eur'
000204 00000:302200 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 Peer timeout reset to 30 seconds (+/- 2)
000205 00000:302211 7F81E97FA700 ALERT S6A me/openair-cn/src/s6a/s6a_task.c:0080 'hss.openair4G.eur' in state 'STATE_OPEN' waiting for next event.
000206 00000:301867 7F81ADFFB700 DEBUE S6A me/openair-cn/src/s6a/s6a_peer.c:0115 Diameter Identity of MME: msi-laptop.openair4G.eur with length: 24
000207 00000:301888 7F81ADFFB700 DEBUE S6A me/openair-cn/src/s6a/s6a_peer.c:0150 S6a peer connection attempt 1 / 8
000208 00000:301905 7F81ADFFB700 DEBUE S6A me/openair-cn/src/s6a/s6a_peer.c:0163 Peer hss.openair4G.eur is now connected...
000209 00000:260339 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0033
000210 00000:260347 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0034
000211 00000:260352 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0036
000212 00000:260356 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0038
000213 00000:260360 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0040
000214 00000:260364 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0042
000215 00000:260368 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0044
000216 00000:260371 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0045
000217 00000:260375 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0046
000218 00000:260379 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0047
000219 00000:260383 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0048
000220 00000:260387 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0049
000221 00000:260391 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0050
000222 00000:260395 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0051
000223 00000:260399 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0052
000224 00000:260403 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0053
000225 00000:260407 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0054
000226 00000:260411 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0055
000227 00000:260415 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0056
000228 00000:260419 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0057
000229 00000:260423 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0058
000230 00000:260427 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0059
000231 00000:260431 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0060
000232 00000:260435 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0061
000233 00000:260439 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0062
000234 00000:260443 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0063
000235 00000:260447 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0064
000236 00000:260451 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0065
000237 00000:260455 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0066
000238 00000:260459 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0067
000239 00000:260463 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0068
000240 00000:260467 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0069
000241 00000:260471 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0070
000242 00000:260475 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0071
000243 00000:260479 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0072
000244 00000:260483 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0073
000245 00000:260487 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0074
000246 00000:260491 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0075
000247 00000:260495 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0076
000248 00000:260499 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0077
000249 00000:260503 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0078
000250 00000:260507 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0079
000251 00000:260511 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0080
000252 00000:260515 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0081
000253 00000:260519 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0082
000254 00000:260523 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0083
000255 00000:260527 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0084
000256 00000:260531 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0085
000257 00000:260535 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0086
000258 00000:260539 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0087
000259 00000:260543 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0088
000260 00000:260547 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0089
000261 00000:260551 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0090
000262 00000:260555 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0091
000263 00000:260559 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0092
000264 00000:260563 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0093
000265 00000:260567 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0094
000266 00000:260571 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0095
000267 00000:260575 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0096
000268 00000:260579 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0097
000269 00000:260583 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0098
000270 00000:260587 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0099
000271 00000:260591 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0100
000272 00000:260595 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0101
000273 00000:260599 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0102
000274 00000:260603 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0103
000275 00000:260607 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0104
000276 00000:260611 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0105
000277 00000:260615 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0106
000278 00000:260619 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0107
000279 00000:260623 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0108
000280 00000:260627 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0109
000281 00000:260631 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0110
000282 00000:260635 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0111
000283 00000:260639 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0112
000284 00000:260643 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0113
000285 00000:260647 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0114
000286 00000:260651 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0115
000287 00000:260655 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0116
000288 00000:260659 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0117
000289 00000:260663 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0118
000290 00000:260667 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0119
000291 00000:260671 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0120
000292 00000:260675 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0121
000293 00000:260679 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0122
000294 00000:260683 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0123
000295 00000:260687 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0124
000296 00000:260691 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0125
000297 00000:260695 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0126
000298 00000:260699 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0127
000299 00000:260703 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0128
000300 00000:260707 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0129
000301 00000:260711 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0130
000302 00000:260715 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0131
000303 00000:260719 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0132
000304 00000:260723 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0133
000305 00000:260727 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0134
000306 00000:260731 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0135
000307 00000:260735 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0136
000308 00000:260739 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0137
000309 00000:260743 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0138
000310 00000:260747 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0139
000311 00000:260751 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0140
000312 00000:260755 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0141
000313 00000:260759 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0142
000314 00000:260763 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0143
000315 00000:260767 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0144
000316 00000:260771 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0145
000317 00000:260775 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0146
000318 00000:260779 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0147
000319 00000:260783 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0148
000320 00000:260787 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0149
000321 00000:260791 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0150
000322 00000:260795 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0151
000323 00000:260799 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0152
000324 00000:260803 7F81EBFFF700 DEBUE MME-AP src/mme_app/mme_app_statistics.c:0153
000325 0000
```


- HSS and MME Connection Setup

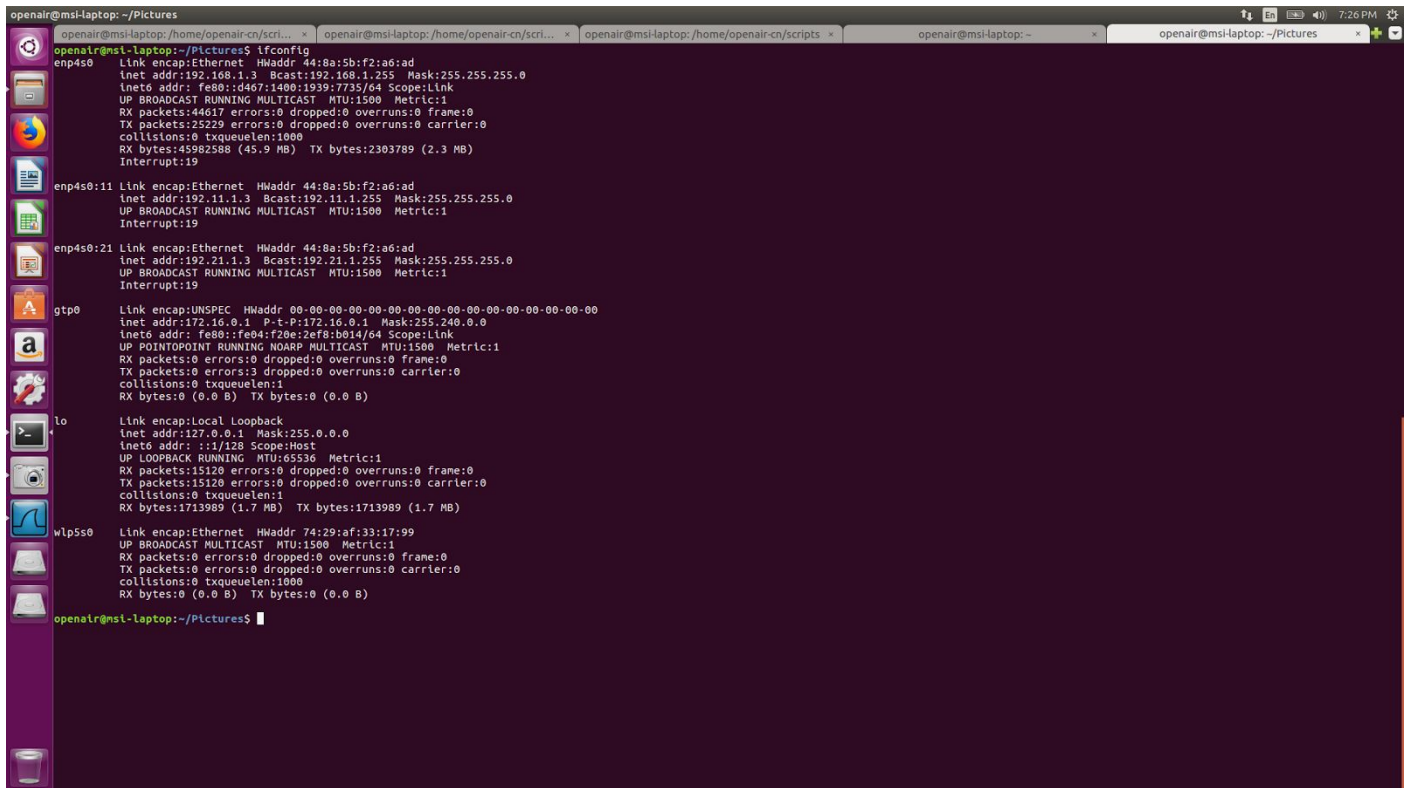
```
opnair@msi-laptop: /home/opnair/cn/scripts ... opnair@msi-laptop: /home/opnair/cn/scripts ... opnair@msi-laptop: /home/opnair/cn/scripts ... opnair@msi-laptop: ~ opnair@msi-laptop: /Desktop

11/27/17, 19:22:28.237329 NOTI
11/27/17, 19:22:28.237333 NOTI
11/27/17, 19:22:28.237336 NOTI
11/27/17, 19:22:28.237341 NOTI
11/27/17, 19:22:28.237343 NOTI
11/27/17, 19:22:28.237345 NOTI
11/27/17, 19:22:28.237347 NOTI
11/27/17, 19:22:28.237351 NOTI
11/27/17, 19:22:28.237353 NOTI
11/27/17, 19:22:28.237358 NOTI
11/27/17, 19:22:28.237361 NOTI
11/27/17, 19:22:28.237364 NOTI
11/27/17, 19:22:28.237367 NOTI
11/27/17, 19:22:28.237372 NOTI
11/27/17, 19:22:28.237376 NOTI
11/27/17, 19:22:28.237378 NOTI
11/27/17, 19:22:28.237390 DBG
11/27/17, 19:22:28.238126 NOTI
11/27/17, 19:22:28.238159 DBG
Initializing ssl layer: DONE
11/27/17, 19:22:43.300876 DBG Incoming connection: {f-----} TCP srv [0.0.0.0]:3868 (6) <- 'localhost' {f-----} TCP from [127.0.0.1]:46988 (6<7))
11/27/17, 19:22:43.301163 DBG RCV from 'unknown': (no mode)0/257 f:R--- src:'(nll)' len:236 [C:264/L:32;C:296/L:21;C:278/L:12;C:257/L:14;C:257/L:14;C:266/L:12;C:269/L:20;C:267/L:12;C:299/
11/27/17, 19:22:43.301208 DBG Created new peer object for incoming CER: msi-laptop.opnair4G.eu
Query: SELECT IMMIDENTITY FROM mneidentity WHERE mneidentity.mnehost='msi-laptop.opnair4G.eu'
Accepting msi-laptop.opnair4G.eu peer
11/27/17, 19:22:43.301601 NOTI Connected to 'msi-laptop.opnair4G.eu' (TCP,soc#7), remote capabilities:
11/27/17, 19:22:43.301615 NOTI 'Capabilities-Exchange-Request'
11/27/17, 19:22:43.301619 NOTI Version: 0x01
11/27/17, 19:22:43.301622 NOTI Length: 236
11/27/17, 19:22:43.301626 NOTI Flags: 0x00 (R---)
11/27/17, 19:22:43.301629 NOTI Command Code: 257
11/27/17, 19:22:43.301632 NOTI ApplicationID: 0
11/27/17, 19:22:43.301638 NOTI Hop-by-Hop Identifier: 0x67C8BFC6
11/27/17, 19:22:43.301647 NOTI End-to-End Identifier: 0x533AA49F
11/27/17, 19:22:43.301650 NOTI {Internal data}: src:(nll)(0) rwb:(nll) rt:0 cb:(nll),(nll)((nll)) qry:(nll) asso:1 sess:(nll)
11/27/17, 19:22:43.301657 NOTI AVP: 'Origin-Host'(264) l=32 f=-M val='msi-laptop.opnair4G.eu'
11/27/17, 19:22:43.301661 NOTI AVP: 'Origin-Realm'(296) l=21 f=-M val='opnair4G.eu'
11/27/17, 19:22:43.301666 NOTI AVP: 'Origin-IP-Address'(278) l=12 f=-M val='192.168.1.3' (0x5a1cac53)
11/27/17, 19:22:43.301669 NOTI AVP: 'Host-IP-Address'(257) l=14 f=-M val='192.11.1.3'
11/27/17, 19:22:43.301674 NOTI AVP: 'Host-IP-Address'(257) l=14 f=-M val='192.21.1.3'
11/27/17, 19:22:43.301677 NOTI AVP: 'Host-IP-Address'(257) l=14 f=-M val='192.168.1.3'
11/27/17, 19:22:43.301682 NOTI AVP: 'Vendor-Id'(266) l=12 f=-M val=0 (0x0)
11/27/17, 19:22:43.301685 NOTI AVP: 'Product-Name'(269) l=20 f=-M val='freediameter'
11/27/17, 19:22:43.301690 NOTI AVP: 'Firmware-Revision'(267) l=12 f=-M val=10200 (0x27d8)
11/27/17, 19:22:43.301695 NOTI AVP: 'Inband-Security-Id'(299) l=12 f=-M val='NO_INBAND_SECURITY' (0 (0x0))
11/27/17, 19:22:43.301700 NOTI AVP: 'Vendor-Specific-Application-Id'(268) l=32 f=-M val=(grouped)
11/27/17, 19:22:43.301704 NOTI AVP: 'Auth-Transaction-Id'(258) l=12 f=-M val='1677251 (0x1000223)'
11/27/17, 19:22:43.301718 NOTI AVP: 'Vendor-Id'(266) l=12 f=-M val='10415 (0x28af)'
11/27/17, 19:22:43.301723 NOTI AVP: 'Supported-Vendor-Id'(265) l=12 f=-M val='10415 (0x28af)'
11/27/17, 19:22:43.301730 NOTI SENT to 'msi-laptop.opnair4G.eu': 'Capabilities-Exchange-Answer' 0/257 f:---- src:'(nll)' len:232 [C:268/L:12;C:264/L:25;C:296/L:21;C:278/L:12;C:257/L:14;C:257/
11/27/17, 19:22:43.301736 L:29;C:267/L:12;C:269/L:20;C:267/L:12;C:299/
NO TLS protection negotiated with peer 'msi-laptop.opnair4G.eu'.
11/27/17, 19:22:43.301797 NOTI 'STATE_CLOSED' -> 'STATE_OPEN' 'msi-laptop.opnair4G.eu'
11/27/17, 19:22:43.301781 DBG SENT to 'msi-laptop.opnair4G.eu': 'Device-Watchdog-Request' 0/280 f:R--- src:'(nll)' len:84 [C:264/L:25;C:296/L:21;C:278/L:12]
11/27/17, 19:22:43.301785 DBG RCV from 'msi-laptop.opnair4G.eu': (no mode)0/280 f:--- src:'msi-laptop.opnair4G.eu' len:160 [C:268/L:12;C:264/L:13;C:296/L:21;C:278/L:12]
11/27/17, 19:22:43.301785 DBG SENT to 'msi-laptop.opnair4G.eu': 'Device-Watchdog-Response' 0/280 f:R--- src:'(nll)' len:84 [C:264/L:25;C:296/L:21;C:278/L:12]
11/27/17, 19:22:43.301791 DBG RCV from 'msi-laptop.opnair4G.eu': (no mode)0/280 f:R--- src:'msi-laptop.opnair4G.eu' len:100 [C:268/L:12;C:264/L:13;C:296/L:21;C:278/L:12]
11/27/17, 19:22:43.301791 DBG RCV from 'msi-laptop.opnair4G.eu': (no mode)0/280 f:R--- src:'msi-laptop.opnair4G.eu' len:100 [C:268/L:12;C:264/L:13;C:296/L:21;C:278/L:12]
11/27/17, 19:22:43.301791 DBG SENT to 'msi-laptop.opnair4G.eu': 'Device-Watchdog-Answer' 0/280 f:R--- src:'(nll)' len:96 [C:268/L:12;C:264/L:25;C:296/L:21;C:278/L:12]
```

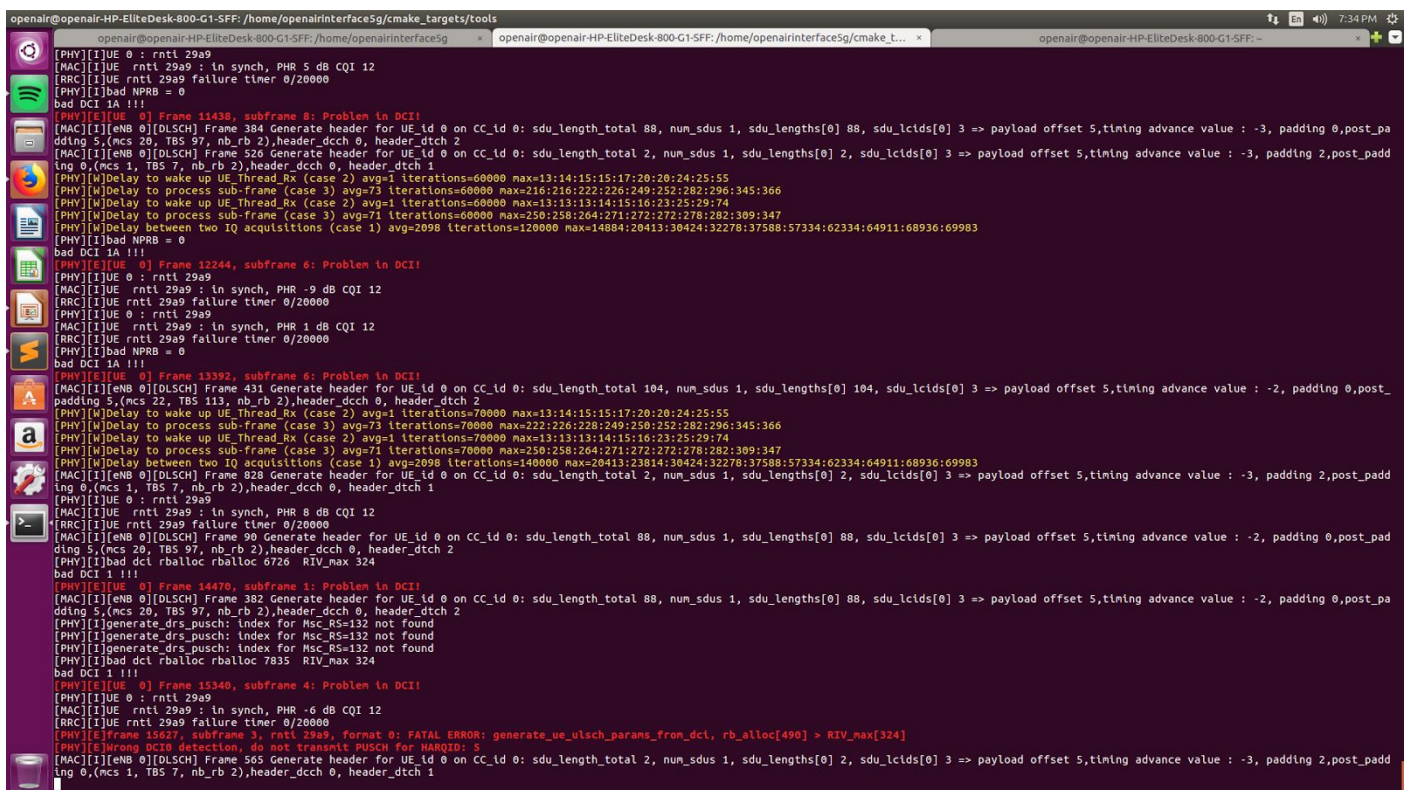
- SPGW Launch

```
openal@msi-laptop: /home/openal/cn/scripts ~$
000090 00000:975158 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0389 55_SB ip (read).....: 192.168.1.3
000091 00000:975165 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0390 55_SB MTU (read).....: 1500
000092 00000:975172 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0391 - SGW
000093 00000:975177 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0392 SGL_iface .....: enp450
000094 00000:975182 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0393 SGL_ip (read).....: 192.168.1.3
000095 00000:975188 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0394 SGL MTU (read).....: 1500
000096 00000:975193 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0396 - MSS clamping: .....: 0
000097 00000:975202 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0397 - Masquerading: .....: 1
000098 00000:975207 7BDCDF3F7700 INFO SPGW-A /openal/cn/src/sw/gpw_cfg.c:0398 - Push PCO: .....: 0
Initializing MSC logs
Initializing MSC logs Done
000099 00000:982990 7BDCDF3F7700 DEBUG UDP /src/udp/udp_primitives_server.c:0356
000100 00000:984126 7BDCDF3F7700 DEBUG UDP /src/udp/udp_primitives_server.c:0364
000101 00000:984142 7BDCDF3F7700 DEBUG S11 ome/openal-cn/src/s11/s11_sgw.c:0259
000102 00000:984147 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0218
000103 00000:984260 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0219
000104 00000:984265 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0220
000105 00000:984269 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0221
000106 00000:984273 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0222
000107 00000:984276 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0223
000108 00000:984281 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0224
000109 00000:984289 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0225
000110 00000:984297 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0226
000111 00000:984304 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0227
000112 00000:984310 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0228
000113 00000:984316 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0229
000114 00000:984320 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0230
000115 00000:984327 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0231
000116 00000:984333 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0232
000117 00000:984340 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0233
000118 00000:984345 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0234
000119 00000:984353 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0235
000120 00000:984359 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0236
000121 00000:984365 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0237
000122 00000:984372 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0238
000123 00000:984377 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0239
000124 00000:984382 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0240
000125 00000:984389 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0241
000126 00000:984394 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0242
000127 00000:984400 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0243
000128 00000:984406 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0244
000129 00000:984413 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0245
000130 00000:984419 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0246
000131 00000:984425 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0247
000132 00000:984431 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0248
000133 00000:984438 7BDCDF3F7700 INFO GTPV2- 2-c/nwgtptv2c-0.11/src/NwGtpv2c.c:0249
000134 00000:984483 7BDCDF3F7700 DEBUG S11 ome/openal-cn/src/s11/s11_sgw.c:0244
000135 00000:984497 7BDCDF3F7700 DEBUG S11 ome/openal-cn/src/s11/s11_sgw.c:0301
000136 00000:984503 7BDCDF3F7700 DEBUG SPGW-A ome/openal-cn/src/sw/gpw_cfg.c:0148
000137 00000:984521 7BDCDF3F7700 DEBUG UDP /src/udp/udp_primitives_server.c:0216
000138 00000:984522 7BDCDF3F7700 DEBUG UDP /src/udp/udp_primitives_server.c:0216
000139 00000:985226 7BDCDF3F7700 DEBUG UDP /src/udp/udp_primitives_server.c:0187
000140 00000:985225 7BDCDF3F7700 DEBUG UDP /src/udp/udp_primitives_server.c:0187
000141 00000:985225 7BDCDF3F7700 NOTICE GTPV1- /cn/src/gtpv1-u/gtp_mod_kernel.c:0184
000142 00001:048050 7BDCDF3F7700 DEBUG GTPV1- /cn/src/gtpv1-u/gtp_mod_kernel.c:0184
000143 00001:048491 7BDCDF3F7700 NOTICE GTPV1- /cn/src/gtpv1-u/gtp_mod_kernel.c:0111
000144 00001:049638 7BDCDF3F7700 DEBUG GTPV1- /cn/src/gtpv1-u/gtpv1u_task.c:0124
000145 00001:009067 7BDCDF3F7700 DEBUG SPGW-A ome/openal-cn/src/sw/gpw_cfg.c:0208
Tx UDP_INIT IP addr 192.21.1.3
Initializing S11 interface: DONE
Initializing SPGW-APP task interface
alt-cn/src/gtpv1-u/gtpv1u_task.c:0148
Creating new listen socket on address 192.21.1.3 and port 2123
Inserting new descriptor for task id, so 31
Received 1 events
In the GTP kernel mode (gen Id is 27)
Setting route to reach UE net 172.16.0.8 via gtp0
GTP kernel configured
Initializing GTPV1U interface: DONE
Initializing SPGW-APP task interface: DONE
```


- EPC Network Interfaces



- OAISIM Launch



- OAISIM Machine Network Interfaces

```

openair@openair-HP-EliteDesk-800-G1-SFF: /home/openairinterface5g
openair@openair-HP-EliteDesk-800-G1-SFF: /home/openairinterface5g$ ifconfig
eno1
Link encap:Ethernet HWaddr 50:65:f3:2a:34:15
inet addr:192.168.1.2 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: fe80::7bb5:537d:43b1:4a8e/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:761454 errors:0 dropped:0 overruns:0 frame:0
TX packets:476981 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueueLen:1000
RX bytes:776607647 (776.6 MB) TX bytes:65042795 (65.0 MB)
Interrupt:20 Memory:f7f00000-f7f20000

enp2s0
Link encap:Ethernet HWaddr a0:36:9f:7d:c4:65
UP BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueueLen:1000
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
Memory:f7d00000-f7dfffff

lo
Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:30155 errors:0 dropped:0 overruns:0 frame:0
TX packets:30155 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueueLen:1
RX bytes:5152469 (5.1 MB) TX bytes:5152469 (5.1 MB)

oip1
Link encap:AMPR NET/ROM HWaddr
inet addr:172.16.0.2 Bcast:172.16.255.255 Mask:255.255.0.0
UP BROADCAST RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueueLen:100
RX bytes:0 (0.0 B) TX bytes:877 (877.0 B)

openair@openair-HP-EliteDesk-800-G1-SFF: /home/openairinterface5g$

```

- UE Internet Access

```

openair@openair-HP-EliteDesk-800-G1-SFF: /home/openairinterface5g
openair@openair-HP-EliteDesk-800-G1-SFF: /home/openairinterface5g$ ping google.com -n 1 -I oip1
PING google.com (172.217.8.206) from 172.16.0.2 oip1: 56(84) bytes of data:
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=1 ttl=50 time=109 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=2 ttl=50 time=50.6 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=3 ttl=50 time=66.2 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=4 ttl=50 time=96.6 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=5 ttl=50 time=61.4 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=6 ttl=50 time=89.2 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=7 ttl=50 time=100 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=8 ttl=50 time=75.4 ms
64 bytes from ord37s09-lin-f14.1e100.net (172.217.8.206): icmp_seq=9 ttl=50 time=57.0 ms

```

• EPC PING Packets Capture

Capturing from enp4s0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Expression...

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	172.16.0.2	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=48/12288, ttl=64 (reply in 4)
2	0.000198321	192.168.1.3	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=48/12288, ttl=64 (reply in 5)
3	0.007717385	172.217.8.206	192.168.1.3	ICMP	98	Echo (ping) reply id=0x1e9a, seq=48/12288, ttl=64 (request in 2)
4	0.007741410	172.217.8.206	172.16.0.2	ICMP	98	Echo (ping) request id=0x1e9a, seq=48/12288, ttl=64 (request in 1)
5	1.050651609	172.16.0.2	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=49/12544, ttl=64 (reply in 6)
6	1.050659891	192.168.1.3	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=49/12544, ttl=64 (reply in 7)
7	1.058412833	172.217.8.206	192.168.1.3	ICMP	98	Echo (ping) reply id=0x1e9a, seq=49/12544, ttl=64 (request in 6)
8	1.058445548	172.217.8.206	172.16.0.2	ICMP	98	Echo (ping) request id=0x1e9a, seq=50/12800, ttl=64 (reply in 8)
9	2.037524150	172.16.0.2	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=50/12800, ttl=64 (reply in 9)
10	2.037572209	192.168.1.3	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=50/12800, ttl=64 (reply in 10)
11	2.045130291	172.217.8.206	192.168.1.3	ICMP	98	Echo (ping) reply id=0x1e9a, seq=50/12800, ttl=64 (request in 10)
12	2.045151513	172.217.8.206	172.16.0.2	ICMP	98	Echo (ping) request id=0x1e9a, seq=51/13056, ttl=64 (reply in 11)
13	2.045151513	192.168.1.2	192.168.1.255	UDP	86	57621 - 57621 Len=44
14	3.015252983	172.16.0.2	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=51/13056, ttl=64 (reply in 12)
15	3.015252983	192.168.1.3	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=51/13056, ttl=64 (reply in 13)
16	3.023515240	172.217.8.206	192.168.1.3	ICMP	98	Echo (ping) reply id=0x1e9a, seq=51/13056, ttl=64 (request in 13)
17	3.023526764	172.217.8.206	172.16.0.2	ICMP	98	Echo (ping) request id=0x1e9a, seq=52/13312, ttl=64 (reply in 14)
18	3.095368381	172.16.0.2	172.16.255.255	ICMP	122	57621 - 57621 Len=44
19	3.096031171	Micro-St_f2:a6:ad	Netgear-ca:ab:76	ARP	42	Who has 192.168.1.1? Tell 192.168.1.3
20	3.096975759	Netgear-ca:ab:76	Micro-St_f2:a6:ad	ARP	60	192.168.1.1 is at 08:02:8e:ce:ab:76
21	4.008773297	172.16.0.2	172.217.8.206	ICMP	98	Echo (ping) request id=0x1e9a, seq=52/13312, ttl=64 (reply in 24)

Frame 1: 134 bytes on wire (1072 bits), 134 bytes captured (1072 bits) on interface 0

Ethernet II, Src: Hewlett-Packard_24:34:15 (50:60:f2:24:34:15), Dst: Micro-St_f2:a6:ad (44:8a:5b:f2:a6:ad)

Internet Protocol Version 4, Src: 172.16.0.2, Dst: 172.217.8.206

User Datagram Protocol, Src Port: 2152, Dst Port: 2152

IPsec Tunneling Protocol

Flags: 0x30
Message Type: T-PDU (0x00)
Length: 84
T-PDU Data: 4500000540b154000400122daac100002acdf908ce000e66f...

Internet Protocol Version 4, Src: 172.16.0.2, Dst: 172.217.8.206

0100 ... = Version: 4
0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 84
Identification: 0xb615 (46813)
Flags: 0x02 (Don't Fragment)
Fragment offset: 0
Time to live: 64
Protocol: ICMP (1)
Header checksum: 0x22da (validation disabled)
[Header checksum status: Unverified]
Source: 172.16.0.2
Destination: 172.217.8.206
[Source GeoIP: Unknown]
[Destination GeoIP: United States, Mountain View, CA, AS15169 Google Inc., 37.419201, -122.057494]

Internet Control Message Protocol

Type: 8 (Echo (ping) request)
Code: 0
Checksum: 0xe66f [correct]
[Checksum Status: Good]
Identifier (BE): 7834 (0x1e9a)
Identifier (LE): 39454 (0x9a1e)
Sequence number (BE): 48 (0x0030)
Sequence number (LE): 12288 (0x3000)
[Response frame: 4]
Timestamp from icmp data: Nov 27, 2017 19:31:47.000000000 EST
[Timestamp from icmp data (relative): 0.294224969 seconds]
Data (48 bytes)

Frame (frame), 134 bytes

Packets: 340 · Displayed: 340 (100.0%)

Profile: Default