Lab 1 - Part II

Due: Monday Oct 21, 2019 (23:59:59 PM)

1. Goal

In this lab, we are going to use MobileInsight to investigate the LTE network protocol in both data plane and control plane. We will closely observe how web latency is created over the LTE network in an indoor environment and on High-Speed Rails (HSR).

2. Mobileinsight Introduction

We will be using the MobileInsight [http://www.mobileinsight.net], a software tool that collects, analyzes and exploits runtime network information from operational cellular networks, for this lab. It exposes protocol messages on both control plane and (below IP) data plane from the 3G/4G chipset.

See more information about MobileInsight on the following websites:

- A tutorial (http://www.mobileinsight.net/tutorials.html)
- Published paper https://www.cs.purdue.edu/homes/chunyi/pubs/mobicom16-li.pdf

3. Instructions

3.1. A LTE/TCP cross layer analysis

Read the lecture slides carefully and get a fundamental understanding of 4G LTE infrastructure. Note that not all the messages mentioned in the lecture slide are observable or collected through MobileInsight. You are expected to know the RRC procedure of data service setup, that is, RRC connection setup request, RRC connection setup and RRC connection complete. If you are interested in learning more regarding RRC procedure and messages used, please refer to 3GPP TS36.331.

For your convenience, we'll provide captured traces for you to analyze. We tether one Android phone (Xiaomi 5s) to one laptop via USB 3.0. We ran a TCP connection using iperf [https://iperf.fr]. We set up a server whose IP address is 222.29.98.58 running **iperf server** (by running `iperf3 –s`). We captured a trace ***iperf_tcp.pcap*** of above-IP network data on the laptop device running **iperf client**. We also captured LTE traces (They have already been parsed to xml format) through MobileInsight on the Android phone.

Note that there exists a delay between the on-chip time of LTE protocol messages (reported by **MobileInsight**) and the system time of TCP pcap traces. You need to align them before analysis. Specifically, for each TCP/IP packet, you should map it to the underlying 4G LTE signaling messages that are responsible for its delivery. IP packets are encapsulated in the data frames over LTE. LTE adds a fixed-sized frame header (2 bytes) to each IP packet. You can use this linear relation between the PDCP packet recorded by MobileInsight and the IP packet logged by **tcpdump** to obtain the timing offset.

Analyze the traces and do the tasks below:

1) Please retrieve the statistical results per message type: (a) Get the number of unique messages and the numbers of each message type and put this result in the report in descending order. (b) Group message by PHY, MAC, RLC, PDCP, RRC, and NAS. Get the numbers of messages per each group and add this result (in a table or plain text) in the report. Learn which group is most popular and answer why. (c) Select any 1-min window that contains the TCP flow and plot the arrival patterns of all the messages in this selected time window, where the X-axis is timestamp and Y is the message ID (you can define a unique ID for each unique message type). Include this result in your report with the start and end time for the selected time window.

2) Timestamp alignment. Calculate the delay between the on-chip time and the system time for this Android phone. (Hints: you can find the pdu size in **LTE_PDCP_DL_Cipher_Data_PDU** messages for downlink packets and **LTE_PDCP_UL_Cipher_Data_PDU** messages for uplink packets).

3) Calculate the two 3-way TCP handshake durations in this trace and put the results into your report. Briefly describe the difference and explain why.

4) Locate the RRC connection setup procedure happened during the first TCP connection establishment. In principle, the RRC connection setup starts with “RRC Connection Setup Request”, followed by “RRC Connection Setup” and “RRC Connection Setup Complete”. Note that their message type id observed via MobileInsight is the same as **LTE_RRC_OTA_Packet** and these messages can be distinguished by their field names. For example,

   <field name=“lte-rrc.rrcConnectionRequest*”> ··· (where * is a wild character)
   <field name=“lte-rrc.rrcConnectionSetup*”> ··· (where * is a wild character)
   <field name=“lte-rrc.rrcConnectionSetupComplete*”> ··· (where * is a wild character)

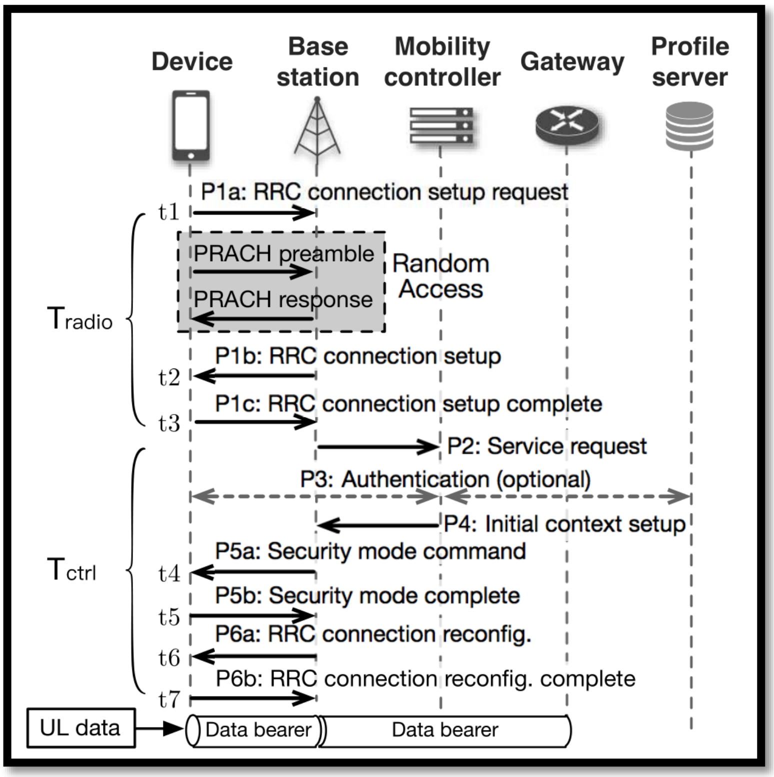

Fig. 1 Control-plane procedures for LTE data access

Please extract these messages and calculate the T_{radio} and T_{ctrl} showed in Fig.1. Please put these results into your report.

3.2. Web latency breakdown

For your convenience, we'll provide captured traces in an indoor environment for you to analyze. To get the traces, we ran Chrome **devtools** on 4 Android phones to capture the 10 websites mentioned in lab1-part I. 3 of the phones were equipped with SIM cards of three mobile carriers in China, while another phone was connected to WIFI. We also ran **tcpdump** to capture the packet-level information on each cell phone. Align the time offset between the **tcpdump** and **MobileInsight** and do the tasks below:

1) Calculate the overall packet retransmission rate at MAC layer and RLC layer, and put the results of 3 different carriers into your report. Choose one web page loading procedure, plot the time series graph. Note that a packet is considered as a packet loss at MAC layer when it fails the CRC check. You can find the CRC results for downlink packets in **LTE_PHY_PDSCH_Stat_Indication** messages and the RLC NACK for downlink packets in **LTE_RLC_UL_AM_All_PDU** messages.
2) Analyze the uplink latency for uplink IP packets. Calculate the mean and standard deviation values of the uplink grant waiting time and the total uplink latency for each packet. Hints:
   a) Each LTE_MAC_UL_Buffer_Status_Internal message contains the uplink data buffer status for the last 40 ms, with each 'UL Buffer Status SubPacket' indicating one millisecond.
   b) LTE_MAC_UL_Transport_Block messages contain the grant information.
3) Based on the processing program you implemented in Lab1-part I, choose 3 page loading procedures (from different websites) and draw visual waterfall graphs for different carriers to show a more detailed breakdown of web latency. Fig. 2 is an example. (Hints: LTE adds a fixed-sized frame header to each IP packet)

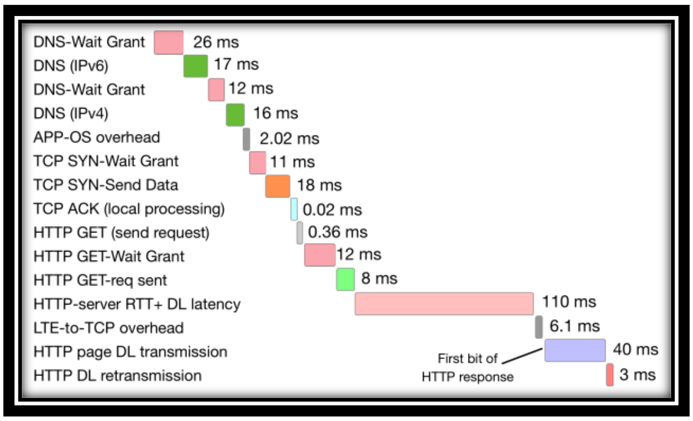

Fig. 2 Data access latency breakdown

4) Compare the page loading over LTE and WIFI network. Describe your observation and explain it.

3.3. Web latency breakdown on HSR

For your convenience, we'll provide traces captured on high-speed rails using the same setting in 3.2. Specifically, one phone was connected to the HSR wifi provided by Fuxing Train. Since 9/2017, China Railway Corporation launched the new "Fuxing Hao" trains for the Beijing-Shanghai HSR route. They bring two notable features: cruising at the speed up to 350 km/h that is faster than any other HSR route in China, and providing free WiFi service via an on-board LTE gateway. The LTE gateway is deployed in an on-train server room by

China Academy of Railway Sciences (CARS). It is equipped with 9 SIM cards of three major Chinese mobile carriers for data relay between LTE RAN and on-board WiFi users.

Analyze the traces and do the tasks below:

1) Calculate the overall packet retransmission rate at MAC layer and RLC layer regarding 3 different carriers. Choose a web page loading process and plot the time series graph. Describe the difference between the indoor environment and high-speed rail scenario.
2) Find handover that happened when a page was loading. For a handover event, analyze the handover duration, data disruption time (i.e. the duration when no data was received), duplicate data transfer time (duplicate data may be detected at PDCP layer). Show the latency breakdown for the handover event you choose. Calculate the overall percentage of page loading that encountered at least one handover.
3) Choose a website and draw visual waterfall graphs for the 4 phones. Compare them with the static ones. What other observations or insights do you have?
4) You are encouraged to collect extra dataset considering different mobility (e.g. walking, taking a taxi, taking a subway, taking a train), or choosing other applications (e.g. videos, instant messages). Note that you need to root your phone and install MobileInsight. Some LTE chips may not be supported by MobileInsight.

4. Project Submission

1. Put all your files into a directory, named "lab1_Name_ID" where ID is your student number.
2. The directory should include the following files:
    a) All the source codes for analysis.
    b) A report of your lab (pdf or word format).
    c) A readme that describes how to run your codes and explains the output files
    d) If you collect your own data, please include the data logs and explain how you collect them and also add the results required into the report.
3. Submit the zipped file containing both parts of lab1 to jing.wang@pku.edu.cn.

Reference

[1] Li, Yuanjie, Zengwen Yuan, and Chunyi Peng. A control-plane perspective on reducing data access latency in lte networks. Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking. ACM, 2017.

[2] Yuan, Zengwen, et al. "A Machine Learning Based Approach to Mobile Network Analysis." 2018 27th International Conference on Computer Communication and Networks (ICCCN). IEEE, 2018.

[3] Tan, Zhaowei, et al. "Supporting mobile VR in LTE networks: How close are we?." Proceedings of the ACM on Measurement and Analysis of Computing Systems 2.1 (2018): 8.

[4] CS590 Lab1: A Closer Look into Data Service Setup Procedure, © Purdue CS590-B5G (PENG)

[5] CS590 Lab2: A Closer Look into Web Latency, © Purdue CS590-B5G (PENG)