MobSys Lab 0: **5G-SA Fundamentals**

Navid Nikaein, Alireza Mohammadi, Chieh-Chun Chen, and Zaineb Benamar

2023-2024

Important Note

While copying text from the PDF, please make sure you respect the spacings. The spaces are not copyable from the PDF, so you should type them manually. For ease of use, a copy of all the code snippets is provided separately in the `code.sh` file.

After one week, a sample file for Wireshark PCAPs would be uploaded to Moodle. Students who cannot finish the lab on time, could use this PCAP to answer the questions at the end of the lab to at least get the grades for the questions.

1 Prerequisites

To interact with the platform you would use the **Command Line Interface (CLI)** binary already installed on your machines. To ease the usage of the CLI, let us create an alias for it and then enable its auto-completion feature:

```
if [ ":$PATH:" != *":/packages/mobsys:"* ]; then
    export PATH="$PATH:/packages/mobsys"
fi
if which cli > /dev/null; then
    source <(cli completion bash)
fi
```

You should add these to your `~/.bashrc+` file to make them permanent. Open a new terminal to make sure that the changes are applied or simply run the following command:

```
$ source ~/.bashrc
```

Then you need to login to the Google Kubernetes Engine (GKE) cluster. First, login to your user account by running the following command:

```
$ gcloud auth login
```

This command will open a browser window and ask you to login to your Google account. If the browser did not open automatically, you can copy the URL from the terminal and paste it in your browser. After logging in, you will be asked to grant access to the Google Cloud SDK.

Now you need to retrieve the credentials for the GKE cluster of your group. To do so, run the following command:

```
$ gcloud container clusters get-credentials ors-cluster{group-id} \
    --region europe-west6 --project comsyslab
```

You should replace the `{group-id}` with your group number.

Now you should be able to use the CLI to interact with the platform. To test it out run the following command:

```
$ cli extract infra
```

In the result you should see a brief description of the infrastructure of the whole cluster, including the different nodes, their properties, and the associated radio components. You should see 3 records.

2 Background

The NG-RAN is divided into control plane (CP) and user plane (UP) as shown in Figure 1. The CP is responsible for the control and management of the radio resources, attachment, and session management. The UP is responsible for the data forwarding.

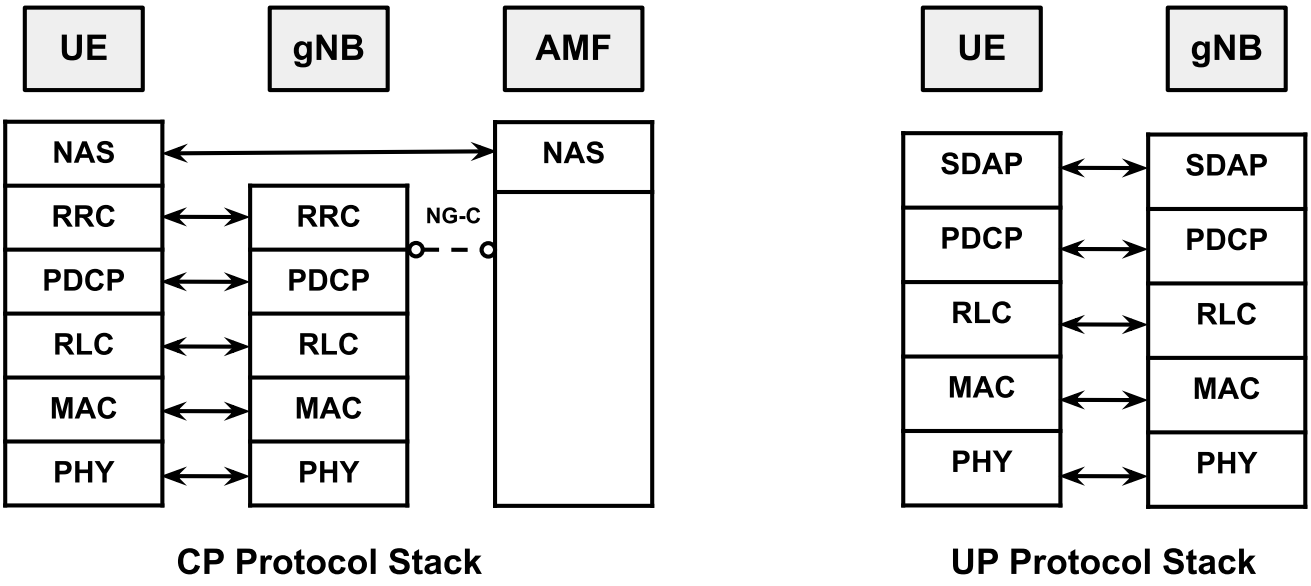

Figure 1: 5G Architecture

5G architecture has two modes:

- Non-standalone (NSA): 5G NSA relies on 4G/LTE for the CP
- Standalone (SA): 5G SA is fully 5G/NR

The mode options are represented by the table 1, where EN-DC refers to Eutra / NR Dual Carrier and NE-DC refers to NR/Eutra Dual Carrier.

Option	SA/NSA	Core	CP-UP	Additional UP	Comment
1	SA	4G-EPC	LTE	-	E-UTRAN
2	SA	5G-CN	NR	-	NR
3/3a/3x	NSA	4G-EPC	LTE	NR	EN-DC
4/4a	NSA	5G-CN	NR	LTE	NE-DC
5	SA	5G-CN	LTE	-	-
6	SA	4G-EPC	NR	-	EPC
7/7a	NSA	5G-CN	LTE	NR	Validation
8/8a	NSA	4G-EPC	NR	LTE	EPC ext: CUPS
?	SA	5G-CN	NR	NR	NR-DC

Table 1: 5G Mode Options

2.1 5G NSA (EN-DC)

In EN-DC, Eutra (LTE) becomes MCG(Main Cell Group) and NR becomes SCG(Secondary Cell Group), meaning that LTE is the main cell and NR just works as a secondary cell. In NE-DC, NR becomes MCG and Eutra (LTE) becomes SCG, meaning that NR is the main cell and LTE just works as a secondary cell. In EN-DC, Core Network is based on LTE Core, NR just provides an additional RAN pipe. When NR is added to the LTE cell, UE should be able to detect SSB of NR cell and perform RACH procedure to NR. Overview of Signaling Procedure: Adding NR Cell to an Existing LTE Cell.

This section focuses on the process of incorporating an NR cell, known as the Secondary Node (SN), into an existing LTE cell, referred to as the Master Node (MN). The signaling flow for this procedure is outlined in the figure 2.

Step 1 : The MN, which is an LTE eNB, initiates the process by sending an SgNB (Secondary gNB) Addition Request to the SN, an NR gNB in this scenario. The LTE eNB conveys several crucial pieces of information to the NR gNB, including:

- Characteristics of the E-RAB.
- Detailed SCG (Secondary Cell Group) configuration information, encompassing the entire UE (User Equipment) capabilities and UE capability coordination results.
- The most recent measurement results for the SN to consider.
- Security information required to enable SRB3 (Signaling Radio Bearer 3).

Step 2 : If the SN decides to accept the request, it responds with an SgNB Addition Request Acknowledgment, performing the following actions:

- Allocating the necessary radio resources and transport network resources.
- Determining the Pscell (Primary Cell) and other SCG Scells (Secondary Cell) and providing the new SCG radio resource configuration to the MN.

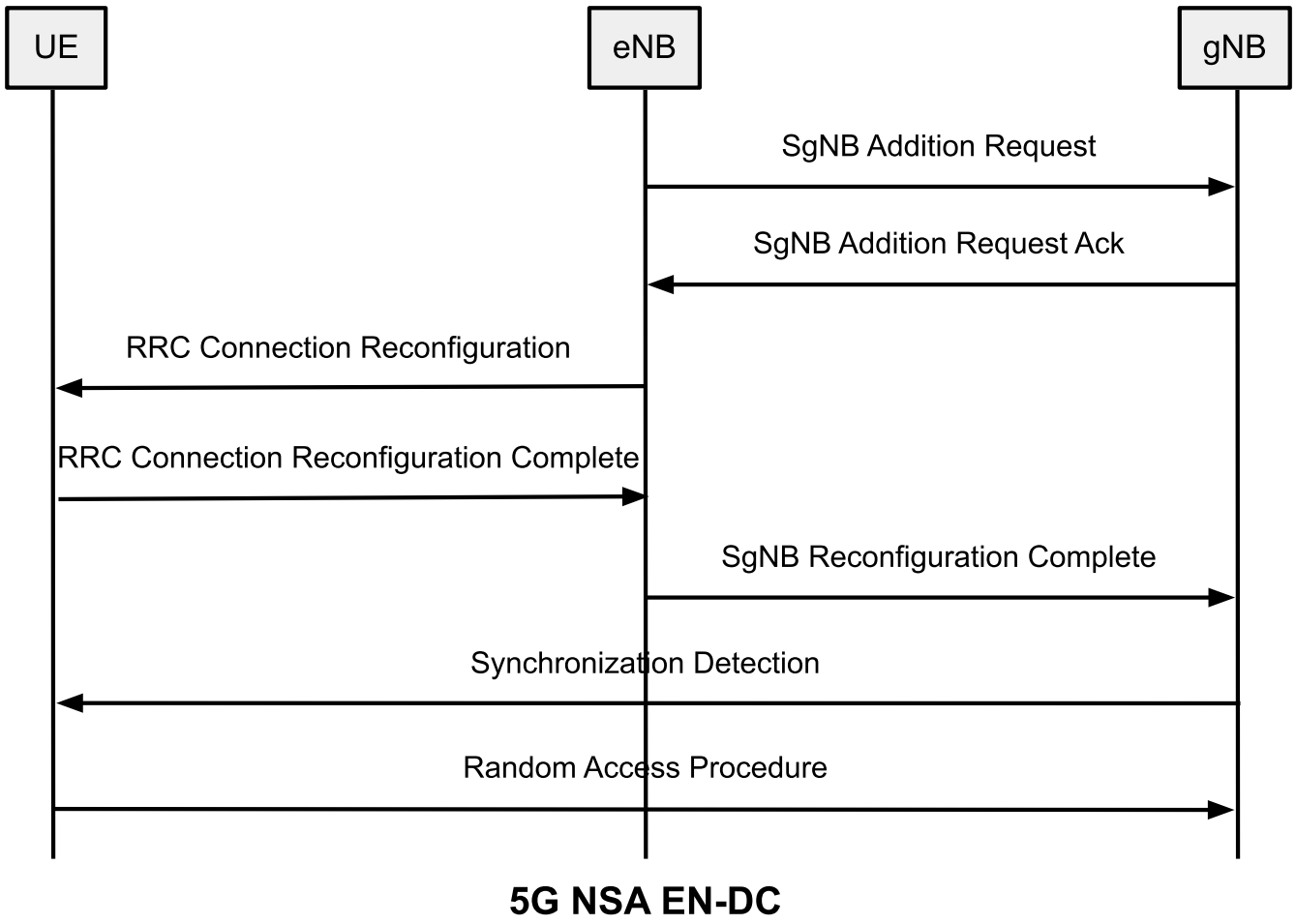

Figure 2: EN-DC Signaling Flow

- In scenarios involving the request for SCG radio resources, offering the SCG radio resource configuration.

Step 3 : Once the NR gNB accepts the SN addition request and provides all the essential information to the LTE eNB, the LTE eNB generates an RRC Connection Reconfiguration message. This message carries NR RRC Connection Configuration information, enabling the UE to determine the necessary configuration details for the NR gNB.

Step 4 : Upon receiving the RRC Connection Reconfiguration message, the UE checks if all the configurations within the message are compatible with the UE's capabilities. If so, the UE sends an RRC Connection Reconfiguration Complete message. This message also includes NR RRC Response data.

Step 5 : Once the LTE eNB (MN) receives the RRC Connection Reconfiguration Complete message from the UE, it informs the NR gNB (SN) that the UE has successfully completed the reconfiguration procedure.

Step 6 : Based on the information contained in the NR RRC Connection Configuration within the RRC Connection Reconfiguration message, the UE detects the synchronization signals block (SSBlock), comprising Primary Synchronization

Signal (PSS), Secondary Synchronization Signal (SSS), and Physical Broadcast Channel (PBCH) of the NR gNB.

Step 7 : After successfully detecting the PSS, SSS, and PBCH of the NR gNB, the UE initiates the Contention-free Random Access Channel (RACH) procedure to connect with the Primary Cell (PSCell) of the Secondary Node (SN or NR gNB). All the necessary information for the RACH procedure is acquired from the RRC Connection Reconfiguration message, eliminating the need for System Information Blocks (SIBs).

2.2 5G SA

Initial Attach is the process that happens when you power on your phone. Following are the procedures that happen at this stage.

1. Scan and synchronize.
2. Receive MIB (Master Information Blocks) and SIB (System Information Blocks).
3. Cell selection and random access (RACH).
4. RRC connection.
5. NAS registration.
6. PDU session establishment.
7. Send/Receive data.

Let us look at each of these steps in more detail:

Step1: Scan and synchronize

- There is no channel dedicated to the UE at the beginning because it is unknown.
- The UE searches for nearby cells and acquires the synchronization signals, Primary Synchronization Signal PSS and
- Secondary Synchronization Signal SSS, to synchronize with the gNB timing.

Step2: Receive MIB and SIB

- SIB and MIB messages are generated by RRC.
- The most important signals that UE has to detect before trying connection in NR SA are MIB and SIB1.
- MIB is carried by the physical channel PBCH.
- PBCH is a part of a SSB.
- SIB1 is carried by the physical channel PDSCH.

- MIB carries information about reference subcarrier spacing, control channel for SIB PDSCH, ...
- SIB1 carries all the basic information for UE to perform the initial attachment procedure and also carries scheduling information for other SIBs.
- There is one major difference between NR and LTE. In LTE, all the SIBs are broadcasted periodically regardless of whether UE wants it or not. However, in NR there are two different types of SIBs. One type is the one being transmitted periodically like SIBs in LTE and the other type is the one being transmitted only when there is a request from UE.

Step3: Cell selection and Contention-Based random access (CBRA) The following messages are exchanged between the UE and the gNB in this step as shown in Figure 3.

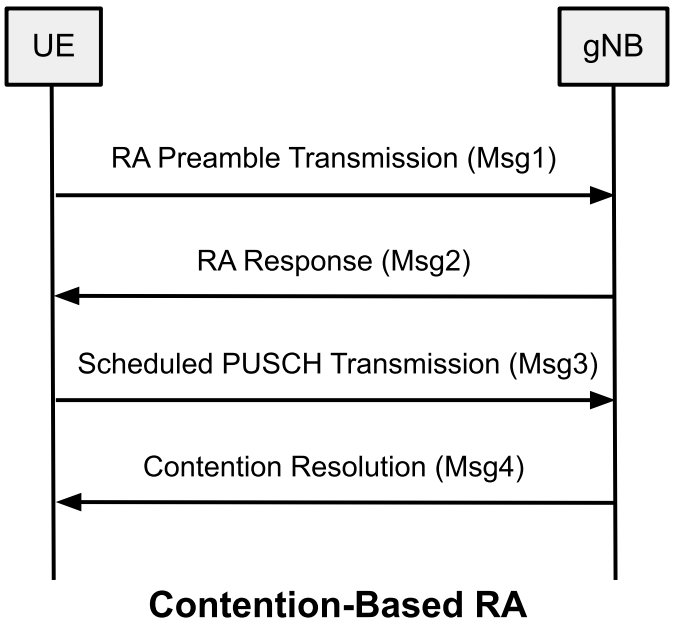

Figure 3: CBRA Signaling Flow

Msg1 (Preamble Transmission): The UE selects a random access preamble from a set of predefined preambles shared with other UEs in the cell and sends it to the gNB on the PRACH.

Msg2 (Random Access Response): Upon receiving Msg1, the gNB sends a random access response called Msg2 on the PDSCH. Msg2 includes several critical pieces of information, such as the Time Advance (TA) command for timing adjustment, the RAPID (Random Access Preamble ID) matching the preamble sent by the UE, and an initial uplink grant for the UE. The gNB also assigns a temporary identifier called TC-RNTI to the UE.

Msg3: After getting the RA response, the UE saves the TC-RNTI and applies the received timing correction. The UE doesn't have a permanent identity so it picks a random number as a UE identity and includes it in msg3 with the TC-RNTI. Using the initial uplink grant provided in Msg2, the UE transmits Msg3 on the PUSCH. Msg3 is a PUSCH which may carry a certain RRC message(e.g, RrcRequest) or just be pure PHY data.

Msg4 (Contention Resolution): After processing Msg3, the gNB sends Msg4 to the UE. Msg4 is a MAC data which is for Contention Resolution. The Contention Resolution message contains the random identity chosen by the UE and is addressed by TC-RNTI, confirming that the gNB has correctly identified the UE, and contention has been resolved. At this step, the network provides UE with C-RNTI which is equal to the TC-RNTI.

In the contention-based random access procedure, the preamble is randomly chosen by the UE from a pool of preambles shared with other UEs in the cell, with the result that more than one UE may transmit the same preamble simultaneously. Hence it will be a collision.

Let's take an example of a collision: Let's say two UEs, UE-A and UE-B, transmit the same PRACH preamble at the same time. In this case, there will be a collision. There are two possibilities:

- gNB is not able to decode preamble sent by any UE: In this case, both UEs will run a backoff timer with some random value and initiate the RA procedure again.
- gNB is able to decode preamble only from UE-A as it has a higher level of power: In this case, gNB will send RAR (msg2) with TC-RNTI for UE-A. Although the RAR is intended for UE-A, both UEs will decode it and work on it because both of them have transmitted the same preamble at the same time. Then, both UEs will choose some random number as an initial identity and send msg3 to the gNB. But gNB will not be able to decode the message from UE-B as UE-B is using the timing advance value that was intended for UE-A. After that, gNB will include the random number of UE-A in msg4 and send it to UE-A. Although both UEs will decode this message because it is addressed by the TC-RNTI, random numbers sent and received by UE-B will mismatch. Only at this stage UE-B will know that it has lost out to some other UE in contention resolution and it should start the RA procedure from the beginning.

Step4: RRC Connection The messages exchanged between the UE and the gNB in this step are shown in Figure 4.

Msg1 (RRC Connection Request): The UE sends an RRC Connection Request message to the gNB. The RRC Connection Request message contains the UE identity, the establishment cause, and the list of the NR frequencies that the UE can measure.

Msg2 (RRC Connection Setup): Upon receiving the RRC Connection Request message, the gNB sends an RRC Connection Setup message to the UE. The RRC Connection Setup message contains the RRC configuration information, including the RRC parameters, the security configuration, and the measurement configuration.

Msg3 (RRC Connection Complete): After receiving the RRC Connection Setup message, the UE sends an RRC Connection Complete message to the gNB. The RRC Connection Complete message contains the UE identity and the RRC parameters.

Msg4 (RRC Connection Reconfiguration): Upon receiving the RRC Connection Complete message, the gNB sends an RRC Connection Reconfiguration message to the UE. The RRC Connection Reconfiguration message contains the RRC parameters and the security configuration.

Msg5 (RRC Connection Reconfiguration Complete): After receiving the RRC Connection Reconfiguration message, the UE sends an RRC Connection Reconfiguration Complete message to the gNB. The RRC Connection Reconfiguration

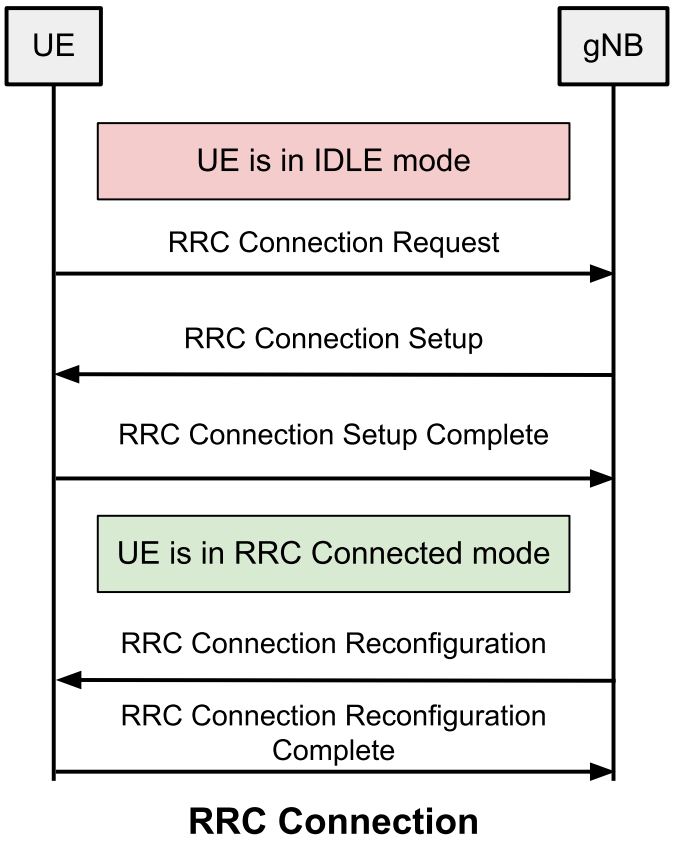

Figure 4: RRC Signaling Flow

Complete message contains the UE identity and the RRC parameters.

Step5: NAS registration Once the RRC connection is established, the UE sends a NAS Registration Request message to the network (specifically the AMF) to register with the 5G network. This message includes important information such as the UE's security credentials and UE's network capability. To ensure secure communication, the network initiates an authentication process. After successful authentication, the UE and network establish security keys for secure communication. Once the security process is complete, the AMF sends a Registration Accept message to the UE, which includes the UE's 5G-GUTI (Globally Unique Temporary Identifier) and other relevant configuration information. The UE is now successfully registered with the 5G network.

Step6: PDU session establishment To establish a data session, the UE sends a PDU (Protocol Data Unit) Session Establishment Request message to the network (specifically the SMF). This message includes the UE's data session requirements, SSC Mode, and the 5G-GUTI. The SMF processes the PDU Session Establishment Request and allocates the necessary resources for the data session. The SMF sends a PDU Session Establishment Accept message to the UE, which contains the PDU session configuration information, such as the allocated QoS and the IP address. The UE is now ready for data transmission over the established PDU session.

Step7: Send/Receive data Finally, the UE is ready to send and receive data.

3 Simple SA Deployment

The first experiment is to deploy a simple 5G Standalone (SA) network using OpenAirInterface (OAI) [1] RF Simulator gNB and OAI minimal 5GC to analyze the control plane and data plane. In this lab we use OAI T-Tracer[2] which traces the protocol stack and messages as well as the radio signals.

Use the following link to configure your Wireshark to decode the OAI T-Tracer messages: https://gitlab.eurecom.fr/oai/openairinterface5g/-/blob/develop/common/utils/T/DOC/T/wireshark.md.

Note

Use the file called `simple-sa.yaml` for this lab.

You need to open the file and replace the PLMN for both the network and the UE to have MCC `001` and MNC of your group number with leading zeros if needed (two digits). Remember to perform the same operation for both the `Network` and the `Terminal` sections (IMSI).

Note

If at any point the deployment fails, refer to the subsection 3.5, for removing the deployment, before trying from the beginning.

3.1 Deployment

Use the command `cli install network sample-sa.yaml` to deploy the network. It should finish without errors and printout the two Kubernetes resource names that were created. Check for the status of the deployment using the command `cli observe`. Wait until all the **Elements** other than the UE are in the `STATUS` set to `1/1 Y` state.

Questions

1. Explain the role of each of the **Elements** in the deployment.
2. Record the output of the observe by taking a screenshot and add it to your report.

Note

The deployment should reach the mentioned state in less than 3 minutes. If it is taking longer, please ask for help.

[1] https://openairinterface.org/
[2] https://gitlab.eurecom.fr/oai/openairinterface5g/-/wikis/T

3.2 PCAP Extraction

We need to extract the PCAP from the gNB to analyze the control plane and data plane traffic. To do so, we specify a set of ports to be captured and then extract the PCAP file. Port 9999 is used by OAI gNB to dump the MAC layer packets internally. The `{element}` is a placeholder for the name of the element. Press the `Tab` key (multiple times) to see the list of available elements and then pick the right one. If you start typing the name of the element, pressing the `Tab` key will complete the name.

```
$ udp_ports="2123 or 2152 or 9999"
$ sctp_ports="38412 or 38472 or 36421"
$ filter="(sctp port $sctp_ports) or (udp port $udp_ports)"
$ cli extract pcap {element} -- "$filter" | wireshark -k -i -
```

OAI gNB is using TTracer which is waiting for the user to call the starting function. On a separate terminal, run the following command to start the gNB:

```
$ cli cic {element} run -- t-dumper
```

Use `Ctrl+C` to stop the command, after you see the line is printed.

Wait for the `cli observe` command to indicate your UEs are ready. To verify the E2E connection, let us check the AMF and gNB logs. To do so, run the following command:

```
$ cli extract logs {element}
```

You could use the `Tab` key to complete the name of the element.

Questions

1. Explain what are each of the ports used for in 4G/5G interfaces and show the protocol stack.
2. How many Logical Channels are created in the gNB? Explain the role of each and connect them to the DRBs.
3. Investigate if there is any user traffic in the PCAP already and if any the type of the traffic (after the UEs readiness).
4. How are we specifying the gNB to wait for the TTracer based on the deployment file?

Note

Continue exercise **WITHOUT** closing the Wireshark.

3.3 Traffic Generation

The next step is to generate traffic between the UE and the gNB and observe the results. Pick one of the UEs at random for this part of the lab. There are two classes of tests that can be performed:

1. `rtt`: Round Trip Time (RTT) measurements using ping.

Direction	Protocol	Command
UL	TCP	`cli test throughput {terminal} ul -- gateway --time 10`
DL	TCP	`cli test throughput {terminal} dl -- gateway --time 10`
	UDP	`cli test throughput {terminal} dl -- gateway --udp --bandwidth 40M`

Table 2: Throughput Test Commands

2. `throughput`: Throughput measurements using iperf3.

In all the following commands of this section, `{terminal}` is a placeholder for the name of the network terminal (a generic term for the UEs). You could use the `Tab` key to see the list of available terminals and then pick the right one. If you start typing the name of the terminal, pressing the `Tab` key will complete the name.

To run the RTT test, use the following command:

```
$ cli test rtt {terminal} -- -c 100 -s 64 12.1.1.1
```

The throughput test could be performed in Uplink (UL) and Downlink (DL) directions with TCP or UDP payloads. Use the table 2 to find the right command for each test.

Questions

1. Run the RTT test and record the statistics.
2. Run the RTT command again by replacing the packet size of 64 bytes with 768 bytes and record the statistics.
3. Compare the two previous results in a bar chart for average, min, max, and standard deviation statistics.
4. Explain where does the IP address 12.1.1.1 in the command come from?
5. Run all the throughput tests and record the statistics.
6. Stop the Wireshark capture and save it to a file for further analysis.
7. Run the TCP downlink tests again and use the option `--plot` before the terminal name to plot the throughput in real-time and record the results by taking a screenshot.

3.4 Extract Configuration

As an extra exercise, you can extract the configuration of the gNB as well as some visuals on your deployment. To extract the configuration, use the following command:

```
$ cli extract config {element} /tmp
```

This will create the configurations in the `/tmp` directory. Keep the configuration for verifying the results of your packet analysis.

1. Find the configuration file for the gNB and verify the PLMN and slice configuration is applied properly.

3.5 Uninstall

To uninstall the network, use the following command:

```
$ cli remove network sample-sa.yaml
```

Checking via the `cli observe` command, you should see that all the elements are removed.

3.6 Questions

For the questions below, when applicable, verify the results both from the configuration or the logs and the PCAP. If one piece of information is present in multiple messages, please specify all of them and explain the differences.

1. What are the IP address of gNB and AMF?
2. What is the port number of N2-AMF?
3. What is the gNB identity? Explain the difference between the gNB ID and the cell ID.
4. What are the frequency and bandwidth of the gNB?
5. What is the subcarrier spacing of the gNB?
6. What TDD pattern is used by the gNB?
7. What preamble index is used for the RACH and what is the type of the RACH?
8. What are the RNTIs of the connected UE?
9. What are the IMSIs and PLMNs of the connected UEs?
10. How did the UEs signal its selected PLMN Identities?
11. Which (logical) channels are used during the RACH and RRC connection setup?
12. Map the RRC connection messages to the logical channels and RLC mode.
13. At which message numbers are the UEs connected to the gNB and can start transmitting data?
14. At which message numbers are the UEs disconnected from the gNB and from AMF?
15. Draw the message sequence chart[3] and show the messages and their associated network entity (i.e. UE, gNB, AMF). You could use the chart generator at `http://mscgen.js.org`. [4]
16. Scheduling Request (SR):
    - Describe a SR with your own words.

[3] Check the time constraint MSC from `https://en.wikipedia.org/wiki/Message_sequence_chart`
[4] To learn how to use MSCGen, refer to `https://mscgen.js.org/tutorial.html`

- How often are SRs received as per the trace?
- What is the SR configuration, i.e., how often can a UE request a SR?

17. By which physical channel is the MIB carried?
18. By which physical channel is the SIB1 carried?
19. What are the periods of the SIB1 and MIB messages?
20. At what message number is the UE connected to the gNB and can start transmitting data?
21. Use the figure 4 and annotate the messages with the time and the message number for each UE.