

# **Course: Introduction to 5G Networks**

## **5G Assignments**

**Team 5:**

**Jaishana Bindhu Priya (21bcs045)**

**Pawar Parth Sanjay (21bcs083)**

**R Eswar Naik (21bec035)**

**D.Rishikesh(21bec013)**

**R Hari prasad  
(21bec018)**

# Assignment 1

## 5G Core Study Report

### Introduction

This report details our experience deploying a 5G core on a server and testing it with a 5G RAN and UE simulator using Ansible Open 5GS. <https://open5gs.org/> The deployment process leverages automation to simplify the setup and provides valuable insights into the functionalities of Open 5GS.

### Open 5GS: Building a Private Network

Open 5GS offers the potential to construct a private NR/LTE network with additional hardware. This section explores the necessary components:

- **gNB/eNB:** These base stations are required for the network. gNB refers to a 5G base station, while eNB refers to a 4G base station.
- **USIM:** These are Universal Subscriber Identity Modules, similar to SIM cards but designed for 5G networks.

### Steps:

- 1) Go within the ansible-opens from github.

```
→ ~ cd Desktop/ansible-open5gs
→ ansible-open5gs git:(main) ✘ ls
inventory.txt  playbook.yml  README.md
→ ansible-open5gs git:(main) ✘
→ ansible-open5gs git:(main) ✘ sudo vi playbook.yml
```

- 2) Change the server value

```
name: Deploy a 5G Core
hosts: 5g_server
become: yes
vars:
  server_value: 192.168.5.95
tasks:
  - name: Update all packages to their latest version
    apt:
      name: "*"
      state: latest

  - name: Install Dependency
    apt:
      name: software-properties-common
      state: present

  - name: Add open5gs latest repository from PPA
    ansible.builtin.apt_repository:
      repo: 'ppa:open5gs/latest'

  - name: Again Update all packages to their latest version
    apt:
      name: "*"
      state: latest

  - name: Install Open5GS
    apt:
      name: open5gs
      state: present

"playbook.yml" 98L, 2300C
```

- 3) Enter the credentials in the .txt file
- 4) Deploy the 5g core

```
ansible-open5gs git:(main) ✘
ansible-open5gs git:(main) ✘ ansible-playbook playbook.yml -i inventory.txt

PLAY [Deploy a 5G Core] ****
ASK [Gathering Facts] ****
```

## Output:

```
PLAY [Deploy a 5G Core] *****
TASK [Gathering Facts] *****
ok: [5g_server]

TASK [Update all packages to their latest version] *****
changed: [5g_server]

TASK [Install Dependency] *****
ok: [5g_server]

TASK [Add open5gs latest repository from PPA] *****
changed: [5g_server]

TASK [Again Update all packages to their latest version] *****
ok: [5g_server]

TASK [Install Open5GS] *****
changed: [5g_server]

TASK [Update AMF config] *****
changed: [5g_server]

TASK [Update UPF config] *****
changed: [5g_server]

TASK [Restart AMF Service] *****
changed: [5g_server]

TASK [Restart UPF Service] *****
changed: [5g_server]

TASK [Restart UPF Service] *****
changed: [5g_server]

TASK [NAT Port Forwarding] *****
changed: [5g_server]

TASK [Update all packages to their latest version] *****
ok: [5g_server]

TASK [Install curl] *****
ok: [5g_server]

TASK [curl package] *****
changed: [5g_server]

TASK [Install nodejs] *****
changed: [5g_server]

TASK [Clone code open5gs] *****
changed: [5g_server]

TASK [Run Dashboard] *****
```

## 5) Run command

```
Steps for open5gs:

# run this command if you are on remote server and want to access dashboard locally
$ ssh -L localhost:3000:localhost:3000 ubuntu@ip
```

Output:

```
→ ansible-openSgs git:(main) ✘ ssh -L localhost:3000:localhost:3000 ubuntu@192.168.5.96
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-91-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 System information as of Wed Apr 13 05:39:58 UTC 2022

 System load:  0.21           Users logged in:      1
 Usage of /:   16.0% of 19.21GB  IPv4 address for eth0:  192.168.5.96
 Memory usage: 10%            IPv4 address for ogstun: 10.45.0.1
 Swap usage:   0%              IPv6 address for ogstun: 2001:db8:cafe::1
 Processes:    175

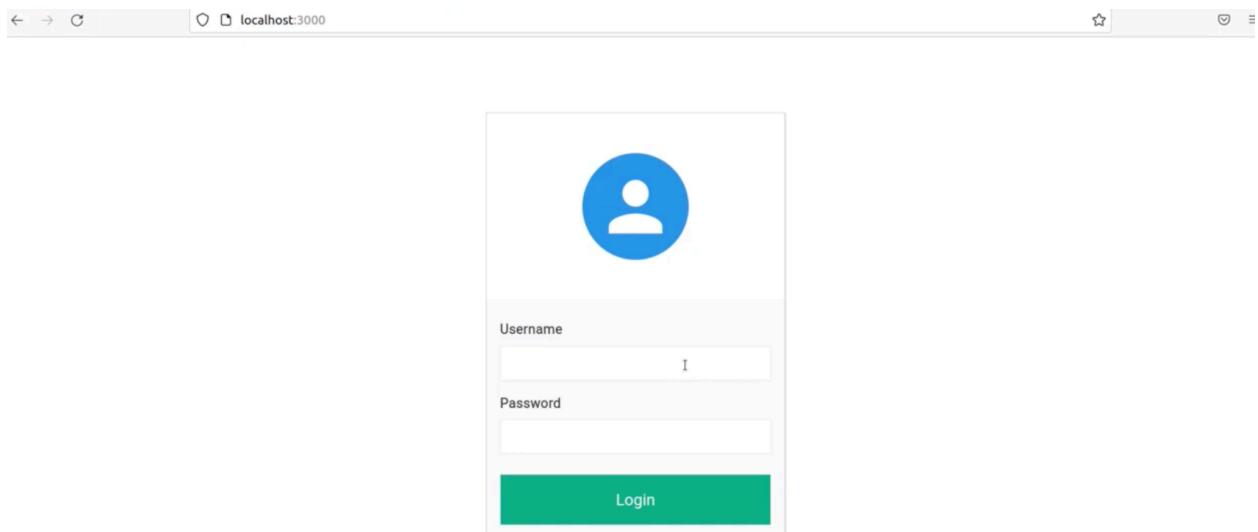
0 updates can be applied immediately.

*** System restart required ***
Last login: Wed Apr 13 05:37:58 2022 from 125.63.65.168
ubuntu@ubuntu:~$ █
```

6) Go to the localhost site- Dashboard

Username: admin

Password: 1234



## 7) Create a subscriber

Create Subscriber

Subscriber Configuration

IMSI\*

90170000000001

Subscriber Key (K)\*

465B5CE8 B199B49F AA5F0A2E E238A6BC

Authentication Management Field (AMF)\*

8000

USIM Type

OPc

Operator Key (OPc/OP)\*

E8ED289D EBA952E4 283B54E8 8E6183CA

UE-AMBR Downlink\*

1 Gbps

UE-AMBR Uplink\*

1 Gbps

CANCEL SAVE

## 8) Check if all services are running using command:

```
buntu@ubuntu:~$ systemctl | grep open5gs
open5gs-amfd.service                                loaded active running  Open5GS AMF Daemon
open5gs-ausfd.service                               loaded active running  Open5GS AUSF Daemon
open5gs-bsfd.service                               loaded active running  Open5GS BSF Daemon
open5gs-hssd.service                               loaded active running  Open5GS HSS Daemon
open5gs-mmed.service                               loaded active running  Open5GS MME Daemon
open5gs-nrfd.service                               loaded active running  Open5GS NRF Daemon
open5gs-nssfd.service                               loaded active running  Open5GS NSSF Daemon
open5gs-pcfd.service                               loaded active running  Open5GS PCF Daemon
open5gs-pcrfd.service                               loaded active running  Open5GS PCRF Daemon
open5gs-sgwcd.service                               loaded active running  Open5GS SGW-C Daemon
open5gs-sgwud.service                               loaded active running  Open5GS SGW-U Daemon
open5gs-smfd.service                               loaded active running  Open5GS SMF Daemon
open5gs-udmd.service                               loaded active running  Open5GS UDM Daemon
open5gs-udrd.service                               loaded active running  Open5GS UDR Daemon
open5gs-upfd.service                               loaded active running  Open5GS UPF Daemon
```

## 9) Install 5g ran and UE simulator - UERANSIM

## Prerequisites

```
# install cmake and other packages
sudo apt update
sudo apt upgrade
sudo apt install iproute2
sudo snap install cmake --classic
sudo apt install gcc
sudo apt install g++
sudo apt install libsctp-dev
sudo apt install make

# clone ueransim
git clone https://github.com/aligungr/UERANSIM
cd UERANSIM
make
```

## 10) Connect with 5g core command: make

```
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CNTypeRestrictionsSet.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CNTypeRestrictionsForS111.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_COUNTValueForPDCP SN111.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_COUNTValueForPDCP SN110.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CPTransportLayerInformation.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CancelAllWarningMessages.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CancelledCellsInEAI_EU.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CanceledCellsInEAI_EU.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CanceledCellsInEAI_NR.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CanceledCellsInEAI_NR.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CanceledCellsInTAI_EU.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CanceledCellsInTAI_EU.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CanceledCellsInTAI_NR.o
[ 5%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CanceledCellsInTAI_NR.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CauseID.c.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CauseMisc.c.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CauseNasLocation.c.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CauseProtocolControlInformation.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CauseRadioNetworkInfo.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CauseTransportInformation.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDBroadcastEUTRAeI.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDBroadcastEUTRAoI.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDBroadcastNR_Identifier.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDBroadcastNR.c.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDCancelledEUTRA_I.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDCancelledEUTRA_o.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDCancelledNR_Identifier.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDCancelledNR.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellIDListForRestart.o
[ 6%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellSize.c.o
[ 7%] Building C object src asn ngap CMakeFiles asn ngap dir ASN NGAP_CellTrafficTraceModule.o
```

```

[ 99%] Built target nr-ue
make[3]: Entering directory '/home/ubuntu/UERANSIM/cmake-build-release'
make[3]: Leaving directory '/home/ubuntu/UERANSIM/cmake-build-release'
make[3]: Entering directory '/home/ubuntu/UERANSIM/cmake-build-release'
[100%] Building CXX object CMakeFiles/devbnd.dir/src/binder.cpp.o
[100%] Linking CXX shared library libdevbnd.so
make[3]: Leaving directory '/home/ubuntu/UERANSIM/cmake-build-release'
[100%] Built target devbnd
make[3]: Entering directory '/home/ubuntu/UERANSIM/cmake-build-release'
make[3]: Leaving directory '/home/ubuntu/UERANSIM/cmake-build-release'
make[3]: Entering directory '/home/ubuntu/UERANSIM/cmake-build-release'
[100%] Building CXX object CMakeFiles/nr-cli.dir/src/cli.cpp.o
[100%] Linking CXX executable nr-cli
make[3]: Leaving directory '/home/ubuntu/UERANSIM/cmake-build-release'
[100%] Built target nr-cli
make[2]: Leaving directory '/home/ubuntu/UERANSIM/cmake-build-release'
make[1]: Leaving directory '/home/ubuntu/UERANSIM/cmake-build-release'
cp cmake-build-release/nr-gnb build/
cp cmake-build-release/nr-ue build/
cp cmake-build-release/nr-cli build/
cp cmake-build-release/libdevbnd.so build/
cp tools/nr-binder build/
JERANSIM successfully built.
Ubuntu@ubuntu:~/UERANSIM$ 
Ubuntu@ubuntu:~/UERANSIM$ 
Ubuntu@ubuntu:~/UERANSIM$ ls
CMakeLists.txt LICENSE README.md build cmake-build-release config makefile src tools
Ubuntu@ubuntu:~/UERANSIM$ cd config/
Ubuntu@ubuntu:~/UERANSIM/config$ ls
custom-gnb.yaml custom-ue.yaml free5gc-gnb.yaml free5gc-ue.yaml open5gs-gnb.yaml open5gs-ue.yaml
Ubuntu@ubuntu:~/UERANSIM/config$ 
Ubuntu@ubuntu:~/UERANSIM/config$ sudo vi open5gs-■

```

## 11) Update Ips

```

mcc: '901'          # Mobile Country Code value
mnc: '70'           # Mobile Network Code value (2 or 3 digits)

nci: '0x000000010' # NR Cell Identity (36-bit)
idLength: 32        # NR gNB ID length in bits [22...32]
tac: 1              # Tracking Area Code

linkIp: 127.0.0.1  # gNB's local IP address for Radio Link Simulation (Usually same with local IP)
ngapIp: 127.0.0.1  # gNB's local IP address for N2 Interface (Usually same with local IP)
gtpIp: 127.0.0.1   # gNB's local IP address for N3 Interface (Usually same with local IP)

# List of AMF address information
amfConfigs:
  - address: 127.0.0.5
    port: 38412

# List of supported S-NSSAIs by this gNB
slices:
  - sst: 1

# Indicates whether or not SCTP stream number errors should be ignored.
ignoreStreamIds: true
~
~
~
~
~
~
~
~
-- INSERT --

```

## 12) Update 5G UE - gnbsearchlist ( provide server id where its deployed)

```

# IMSI number of the UE. IMSI = [MCC|MNC|MSISDN] (In total 15 digits)
supi: 'imsi-9017000000000001'
# Mobile Country Code value of HPLMN
mcc: '901'
# Mobile Network Code value of HPLMN (2 or 3 digits)
mnc: '70'

# Permanent subscription key
key: '465B5CE8B199B49FAA5F0A2EE238A6BC'
# Operator code (OP or OPC) of the UE
op: 'E8ED289DEBA952E4283B54E88E6183CA'
# This value specifies the OP type and it can be either 'OP' or 'OPC'
opType: 'OPC'
# Authentication Management Field (AMF) value
amf: '8000'
# IMEI number of the device. It is used if no SUPI is provided
imei: '356938035643803'
# IMEISV number of the device. It is used if no SUPI and IMEI is provided
imeiSv: '4370816125816151'

# List of gNB IP addresses for Radio Link Simulation
gnbSearchList:
  - 192.168.5.97

# UAC Access Identities Configuration
uacAic:
  mps: false
  mcs: false

# UAC Access Control Class
uacAcc:
-- INSERT --

```

- 13) Run gnodeB, check if its establishing connection with 5g core or not. Run UE parallelly, and in another gnodeB

```

ubuntu@ubuntu:~/UERANSIM$ 
ubuntu@ubuntu:~/UERANSIM$ sudo ./build/nr-gnb -c config/open5gs-gnb.yaml
UERANSIM v3.2.6
2022-04-13 05:58:40.344] [sctp] [info] Trying to establish SCTP connection... (192.168.5.96:38412)
2022-04-13 05:58:40.359] [sctp] [info] SCTP connection established (192.168.5.96:38412)  I
2022-04-13 05:58:40.359] [sctp] [debug] SCTP association setup ascId[3]
2022-04-13 05:58:40.359] [ngap] [debug] Sending NG Setup Request
2022-04-13 05:58:40.359] [ngap] [debug] NG Setup Response received
2022-04-13 05:58:40.359] [ngap] [info] NG Setup procedure is successful

```

14) Connection established. Check if PDU Session is created for subscriber

```
ubuntu@ubuntu:~/UERANSIM$ sudo ./build/nr-ue -c config/open5gs-ue.yaml
JERANSIM v3.2.6
[2022-04-13 05:59:30.154] [nas] [info] UE switches to state [MM-Deregistered/PLMN-SEARCH]
[2022-04-13 05:59:30.155] [rrc] [debug] New signal detected for cell[1], total [1] cells in coverage
[2022-04-13 05:59:30.155] [nas] [info] Selected plmn[901/70]
[2022-04-13 05:59:30.155] [rrc] [info] Selected cell plmn[901/70] tac[1] category[SUITABLE]
[2022-04-13 05:59:30.155] [nas] [info] UE switches to state [MM-Deregistered/PS]
[2022-04-13 05:59:30.155] [nas] [info] UE switches to state [MM-Deregistered/NORMAL-SERVICE]
[2022-04-13 05:59:30.155] [nas] [debug] Initial registration required due to [MM-Dereg-Normal-Service]
[2022-04-13 05:59:30.156] [nas] [debug] UAC access attempt is allowed for identity[0], category[MO_sig]
[2022-04-13 05:59:30.156] [nas] [debug] Sending Initial Registration
[2022-04-13 05:59:30.156] [nas] [info] UE switches to state [MM-REGISTER-INITIATED]
[2022-04-13 05:59:30.156] [rrc] [debug] Sending RRC Setup Request
[2022-04-13 05:59:30.156] [rrc] [info] RRC connection established
[2022-04-13 05:59:30.156] [rrc] [info] UE switches to state [RRC-CONNECTED]
[2022-04-13 05:59:30.156] [nas] [info] UE switches to state [CM-CONNECTED]
[2022-04-13 05:59:30.162] [nas] [debug] Authentication Request received
[2022-04-13 05:59:30.162] [nas] [debug] Sending Authentication Failure due to SQN out of range
[2022-04-13 05:59:30.164] [nas] [debug] Authentication Request received
[2022-04-13 05:59:30.166] [nas] [debug] Security Mode Command received
[2022-04-13 05:59:30.166] [nas] [debug] Selected integrity[2] ciphering[0]
[2022-04-13 05:59:30.173] [nas] [debug] Registration accept received
[2022-04-13 05:59:30.173] [nas] [info] UE switches to state [MM-REGISTERED/NORMAL-SERVICE]
[2022-04-13 05:59:30.173] [nas] [debug] Sending Registration Complete
[2022-04-13 05:59:30.173] [nas] [info] Initial Registration is successful
[2022-04-13 05:59:30.173] [nas] [debug] Sending PDU Session Establishment Request
[2022-04-13 05:59:30.173] [nas] [debug] UAC access attempt is allowed for identity[0], category[MO_sig]
[2022-04-13 05:59:30.379] [nas] [debug] Configuration Update Command received
[2022-04-13 05:59:30.388] [nas] [debug] PDU Session Establishment Accept received
[2022-04-13 05:59:30.389] [nas] [info] PDU Session establishment is successful PSI[1]
[2022-04-13 05:59:30.406] [app] [info] Connection setup for PDU session[1] is successful, TUN interface[uesimtun0, 10.45.0.
```

15)PDU session successfully created!

## Open5GS Architecture

Open 5GS employs a Service-Based Architecture (SBA) for the 5G Standalone (SA) core, contrasting with the traditional architecture used in 4G cores. In an SBA, control plane functions register with the Network Repository Function (NRF), which facilitates discovery of other core functions. Here's a breakdown of the key functions within the

Open5GS 5G SA Core:

- NRF (NF Repository Function): Manages the lifecycle of network functions.
- SCP (Service Communication Proxy): Enables indirect communication between core functions.
- SEPP (Security Edge Protection Proxy): Enhances roaming security.
- AMF (Access and Mobility Management Function): Handles connection and mobility management for user equipment (UE).
- SMF (Session Management Function): Manages data sessions for UEs.

- UPF (User Plane Function): Forwards user data packets between the gNB and external networks.
- AUSF (Authentication Server Function): Generates SIM authentication vectors.
- UDM (Unified Data Management): Manages subscriber data.
- UDR (Unified Data Repository): Stores subscriber data.
- PCF (Policy and Charging Function): Enforces policies and performs charging for subscribers.
- NSSF (Network Slice Selection Function): Assists in network slice selection.
- BSF (Binding Support Function): (Optional) Provides support for network attachment procedures.

### **Key Differences between 4G and 5G Core Architectures**

The 5G SA core user plane is significantly simpler than its 4G counterpart, consisting solely of the UPF. In contrast, the 4G core utilizes multiple functions for user plane management.

### **Configuration Considerations**

With the exception of the SMF and UPF, configuration files for the 5G SA core functions primarily specify the function's IP address binding information and the NRF's IP address or DNS name.