दूरसंचार विभाग
DEPARTMENT OF TELECOMMUNICATIONS

5G LAB BOOK

EXPERIMENTS IN 5G CORE
5G NEW RADIO AND USE CASES

Under Guidance of **Shri Ashok Kumar**
OSD to Sec (T), DDG (SRI) & DDG (Technology), UPW, DoT

Prepared by
Sanjay K Madhukar
ADG (Tech/R2)
UPW LSA, DoT

Edition 1.0

राधेश्याम परमार, भा.दू.से
अपर महानिदेशक, दूरसंचार

Radheshyam Parmar, ITS
Additional Director General, Telecom

सत्यमेव जयते

भारत सरकार,
संचार मंत्रालय
दूरसंचार विभाग
GOVERNMENT OF INDIA
MINISTRY OF COMMUNICATIONS
DEPARTMENT OF TELECOMMUNICATIONS

Following the visionary announcement by the Hon'ble Prime Minister of India regarding the establishment of 100 5G Use Case Labs, the Department of Telecommunications (DoT) has successfully set up these labs across Higher Educational Institutions in every State and Union Territory. Notably, five such labs have been installed in the UP West LSA area, marking a transformative step toward fostering innovation and research in 5G technology.

These labs serve as dynamic hubs for experimentation and development of 5G applications across diverse socioeconomic sectors. Faculty members and students are among the first to benefit from this initiative, gaining hands-on experience and technical expertise in cutting-edge 5G domains. The labs are poised to become centers of excellence, guiding stakeholders in harnessing the full potential of 5G.

I extend my heartfelt appreciation to Shri Ashok Kumar, DDG (Technology), and Shri Sanjay Kr Madhukar, ADG (Technical/R2), for their commendable efforts in compiling and releasing the 5G Lab Book. This comprehensive resource will be invaluable to students and faculty, empowering them to design and develop innovative products and applications using state-of-the-art 5G technology.

I wish continued success to the entire LSA team, and to all students and faculty members engaged in this pioneering journey. May these endeavours lead to groundbreaking contributions that shape the future of digital India.

01.10.25
(Radheshyam Parmar)

अशोक कुमार, भादू..से.
उप महानिदेशक (एस आर आई)
Ashok Kumar, ITS
Deputy Director General (S R I)

सत्यमेव जयते

भारत सरकार
संचार मंत्रालय
दूरसंचार विभाग
Government of India
Ministry of communications
Department of Telecommunications
Website www.dot.gov.in

I am delighted to see the 5G Lab Book, comprehensively detailing 5G Features, the 5G Next Generation Core (NGC), 5G New Radio (NR), and a range of Use Cases for various sectors of Economy, now in the hands of faculty and students. Following the successful installation and commissioning of 5G Use Case Labs, and the completion of User Acceptance Testing (UAT), the need for such a structured documentation of 5G experiments became both urgent and evident.

This Lab Book is thoughtfully designed to guide students and researchers through hands-on explorations in the dynamic realm of 5G technology. It offers a practical pathway to understanding key components of 5G—NGC, NR, and Use Cases—through experiments that bridge theoretical concepts with real-world applications. Each module encourages learners to simulate scenarios and grasp the intricate interplay between radio access and core network functions.

I urge faculty members and researchers across institutions to treat this Lab Book as a foundational resource—one that invites further enrichment through their own contributions of experiments and contextual use cases. May it serve not only as a reference but also as a springboard for deeper engagement with the future of wireless connectivity.

I extend my heartfelt congratulations to UPW LSA and in particular Shri Sanjay Kr Madhukar, ADG at UPW LSA, faculty members from JIIT Noida, IIT Roorkee, GBPUAT, Pantnagar, THDC-IHET Tehri for these commendable initiatives and wish continued success in all future endeavours.

Ashok Kumar
DDG (SRI)
Mob: +91 9818655056
Mail: Ashok.kr100@gov.in

Executive Summary

Subsequent to the successful commissioning of 5G Use case lab in the Institutes, a formal documentation of 5G use case experiments was much needed. This document is prepared to fulfil the much-awaited key requirement and a highly sought-after documentation by the Faculty of Institution of 5G Use Case Lab.

The 5G mobile technology is known for its features of very high-speed, low latency, massive MIMO as well as separation of network in the form of Network Slicing. The Part -1 of this document describes experiments which explores features of 5G mobile technology.

The 5G core is also known for encompassing all the functional module of NG Core and confirms to standards and specification of NG core. The Part-2 of this document takes exploration of captured messages to understand the Call Flow as well as architecture of the 5G core.

The Radio part of the 5G network completes the essential functionality of 5G Technology and gives understanding of Radio communication of access network as well as insights of spectrum aspect. The part-3 of this document takes a number of experiments to explain the Radio Access Network as well as the call set up in the 5G mobile technology.

The concluding part -4 of this document is compilation of 5G use case experiments to be performed with the provided use case equipment such as ANUBHAV Board, IoT Sensors, AI Camera and 5G Drone.

The content of this 5G Lab Book is not exhaustive but is suggestive of practical aspects of the academic insights of the 5G Technology.

Disclaimer

The information presented in this book is intended for educational and informational purposes only. While every effort has been made to ensure the accuracy and reliability of the content, the author makes no representations or warranties regarding the completeness, suitability or applicability of any experimental procedures, technical specifications or outcomes described herein.

Readers are advised to consult with qualified faculty or professionals and adhere to local regulations, safety standards before attempting any 5G-related experiments. The author disclaims any liability for direct or indirect damages resulting from the use or misuse of the information contained in this book.

All trademarks, service marks and product names mentioned are the property of their respective owners and are used for identification purposes only. Any resemblance to actual networks, systems or organizations is purely coincidental unless explicitly stated.

5G Lab Book

INDEX

Part -1: 5G Features

Part -2: 5G NG Core

Protocol Deployment on NG Core Server

Checking Service Status and Functionality of

1. **AMF – Access and Mobility Management Function**
2. **AUSF – Authentication Server Function**
3. **DN – Data Network**
4. **NEF – Network Exposure Function**
5. **NSSF – Network Slice Selection Function**
6. **PCF – Policy Control Function**
7. **SMF – Session Management Function**
8. **UDM – Unified Data Management**
9. **UDR – User Data Repository**
10. **UPF – User Plane Function**
11. **NWDAF – Network Data Analytics Function**

Part -3: 5G NG Radio

1. **Evaluation of gNB Support for Radio Resource Management Functions**
2. **Assessment of gNB Capabilities in Radio Bearer Control**
3. **Verification of gNB Functionality in Radio Admission Control**
4. **Testing gNB Support for Connection Mobility Control Mechanisms**
5. **Analysis of gNB Dynamic Resource Allocation Efficiency**
6. **Validation of gNB Routing of User Plane Data to UPF(s)**
7. **Verification of gNB Routing of Control Plane Information to AMF**
8. **Experiment on gNB Connection Setup and Release Procedures**
9. **Evaluation of gNB Paging Message Scheduling and Transmission**
10. **Assessment of gNB Transport-Level Packet Marking in Uplink Direction**
11. **Testing gNB Session Management Capabilities in 5G Standalone (SA)**

12. **Verification of QoS Flow Management and Mapping to Data Radio Bearers by gNB**

13. **To verify gNB support to paging functionality in 5G SA network**

14. **To verify a successful Registration of UE**

15. **To find PDU Session Working**

Part -4: 5G Use cases

1. 5G Evaluation Board: ANUBHAV

Using Evaluation Board for Use Cases in Smart Homes, Smart City etc.

Automating 5G Evaluation Board Configuration with Bash Scripts

2. IoT Sensor (Coral Gyan)

Coral Gyan Use Case with 5G Evaluation Borad (Anubhav)

IoT Use Cases and Deployment across key Sectors

Example set up utilizing the equipment in 5G Lab for Agriculture sector

Data flow example

3. 5G AI Camera

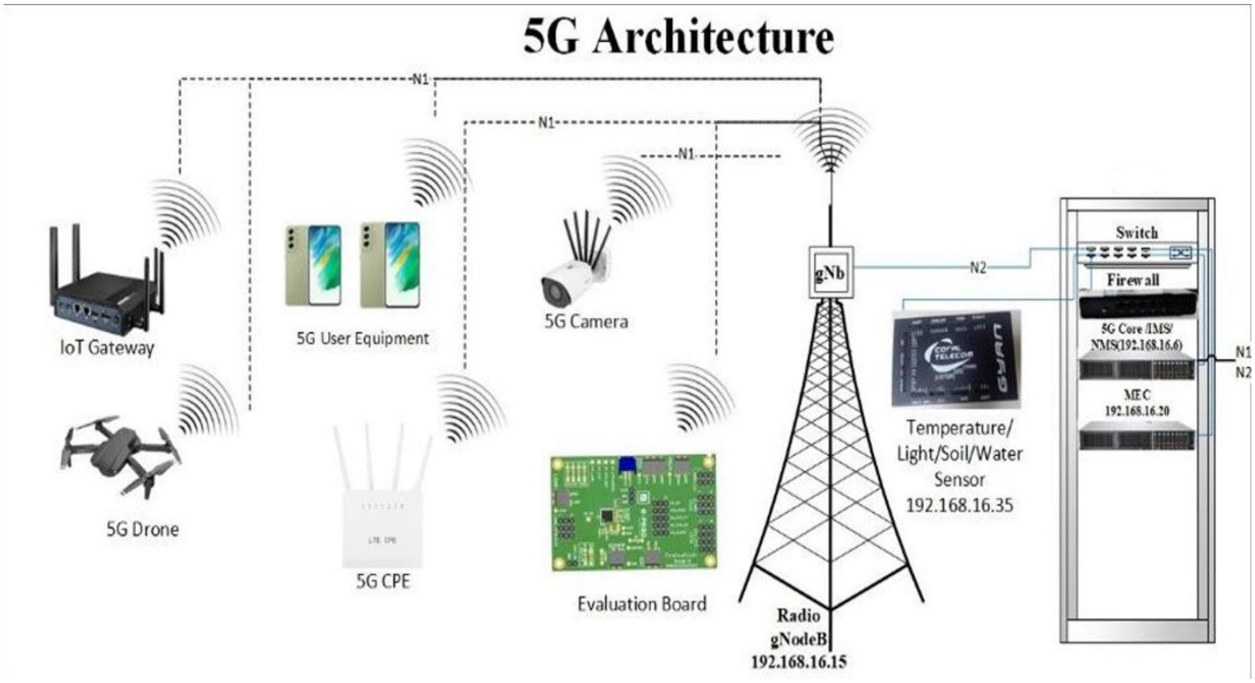
Real-Time Fire Detection Using a 5G AI Camera

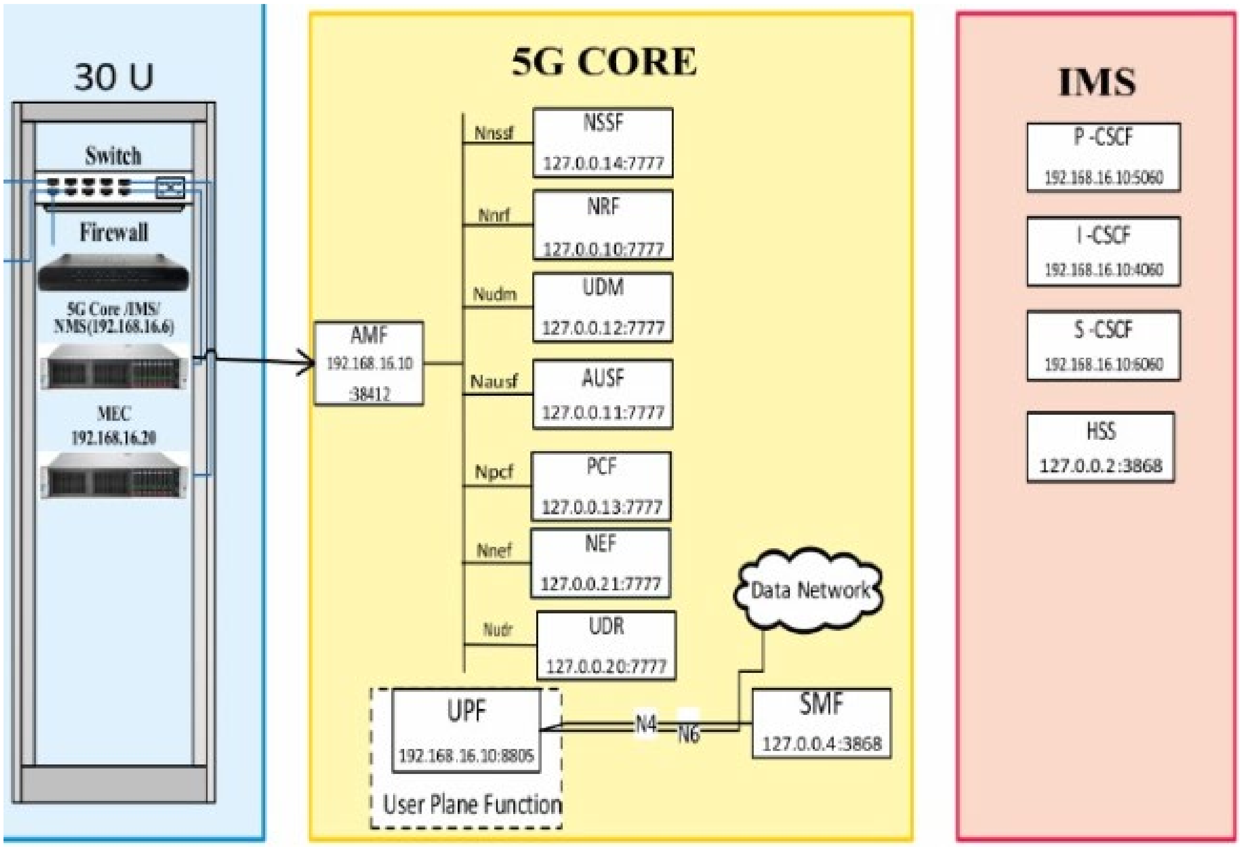
Restricted Access Control System Using a 5G AI Camera

4. 5G Mini Drone

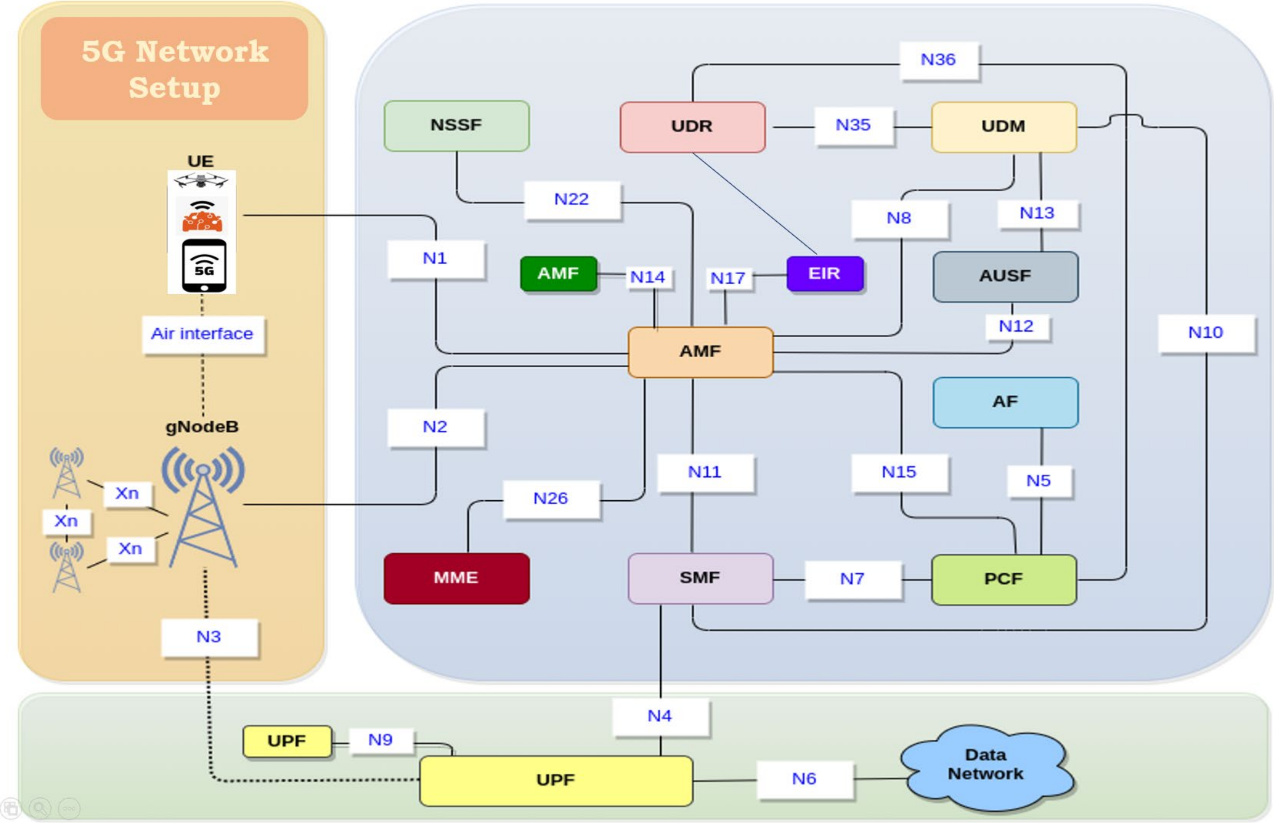
Real-Time Fire Detection Using a 5G Drone

Real-Time Human Detection Using a 5G Drone


5G Architecture
N1
N1
N1
IoT Gateway
5G User Equipment
5G Camera
gNb
N2
Switch
Firewall
5G Core /IMS/ NMS(192.168.16.6)
MEC 192.168.16.20
N1
N2
5G Drone
5G CPE
Evaluation Board
Temperature/ Light/Soil/Water Sensor 192.168.16.35
Radio gNodeB 192.168.16.15

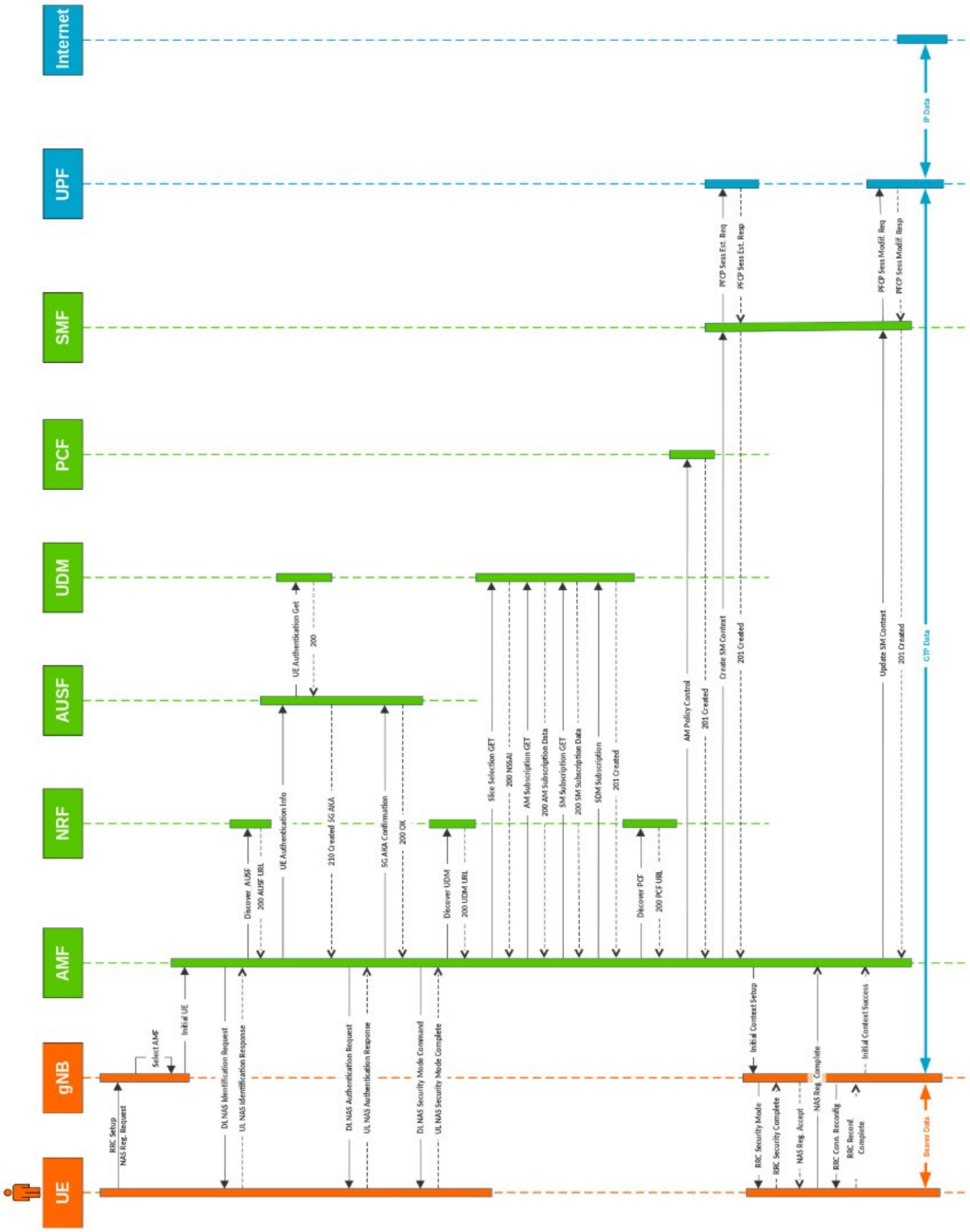

30 U
Switch
Firewall
5G Core /IMS/ NMS(192.168.16.6)
MEC 192.168.16.20
5G CORE
AMF 192.168.16.10 :38412
Nnssf
NSSF 127.0.0.14:7777
Nnrf
NRF 127.0.0.10:7777
Nudm
UDM 127.0.0.12:7777
Nausf
AUSF 127.0.0.11:7777
Npcf
PCF 127.0.0.13:7777
Nnef
NEF 127.0.0.21:7777
Nudr
UDR 127.0.0.20:7777
Data Network
UPF 192.168.16.10:8805
User Plane Function
N4
N6
SMF 127.0.0.4:3868
IMS
P-CSCF 192.168.16.10:5060
I-CSCF 192.168.16.10:4060
S-CSCF 192.168.16.10:6060
HSS 127.0.0.2:3868

Next Generation Core (NGC)



Interface Name	Description	Connected Nodes
N1	Between User Equipment (UE) and Access and Mobility Management Function (AMF). Handles non-radio signalling (NAS)	UE and AMF
N2	Between gNB (Next-Generation NodeB) and AMF	gNB and AMF
N3	Between gNB and User Plane Function (UPF)	gNB and UPF
N4	Between Control Plane (CP) elements (e.g., SMF) and UPF	SMF and UPF
N5	Between User Data Management (UDM) & (PCF)	UDM and PCF
N7	Between Policy Control Function (PCF) and Application Function (AF)	PCF and AF
N11	Between AMF and Session Management Function (SMF)	AMF and SMF
N14	Between AMF and Network Slice Selection Function (NSSF)	AMF and NSSF
N15	Between AMF and Authentication Server Function (AUSF)	AMF and AUSF
N17	Between AMF and Unified Data Repository (UDR)	AMF and UDR
N22	Between AMF and NWDAF (Network Data Analytics Function)	AMF and NWDAF
N26	Enables interworking between 5G and 4G networks (eNB/gNB)	5G gNB and 4G eNB

5G Standalone Call flow



*Referred from open source

Simplified 5G Standalone (SA) registration and call flow

1. Initial access and RRC connection establishment

- **UE searches for network**: The phone or UE looks for a nearby 5G tower (gNB) and sends a signal to say, “I want to connect.”

  i.e. **Random-Access Initiation**: The UE searches for a gNB (5G base station) and initiates a random-access procedure (RAP) to establish initial communication. This includes sending a preamble via PRACH.
- **gNB responds**: The tower hears the signal and gives the phone or UE a temporary channel to talk further.

  i.e. **Resource Allocation by gNB**: The gNB receives the UE's preamble, responds with Random Access Response (RAR) allocating uplink resources and timing advance for the UE to transmit further information.
- **Basic communication starts**: They set up a basic link Signalling Radio Bearer 0 (SRB0) to exchange control messages. UE establishes SRB0

  i.e. **SRB0 Setup**: UE and gNB establish SRB0 for basic RRC and NAS signalling.
- **Connection request**: The phone formally asks to connect (RRC Connection Request), and the tower replies with setup instructions (RRC Setup), upgrading the link to SRB1 for richer communication.

  i.e. **RRC Connection Setup**: UE sends RRC Connection Request. gNB replies with RRC Connection Setup. UE completes with RRC Connection Setup Complete, enabling SRB1 for enhanced signalling.

2. Registration procedure (NAS Layer)

- **Phone introduces itself**: The UE sends a Registration Request to the tower, which passes it to the brain of the network (AMF). i.e. **Initial NAS Registration**: UE sends Registration Request via NAS over SRB1. The gNB forwards it to AMF.
- **Context handover (if needed)**: If the phone was connected to a different AMF before, the new one fetches its history.
  i.e. **AMF Context Handling**: If UE is handed over from another AMF, new AMF initiates context transfer from old AMF.
- **Authentication and Security Set up:**
  - The AMF asks the phone to prove its identity.
    i.e. UE Identity Transfer: The AMF requests the UE's identity for authentication.
  - It checks with AUSF and UDM to verify the phone using secure credentials.
    i.e. Authentication Request: The AMF initiates authentication with the AUSF (Authentication Server Function), which interacts with the UDM (Unified Data Management) to obtain authentication vectors and authenticate the UE.
  - First, the phone and AMF agree on encryption rules for NAS-level messages.
    i.e. Security Mode Command and Complete: The AMF and UE establish NAS (Non-Access Stratum) level security (ciphering and integrity protection) through Security Mode Command and Security Mode Complete messages.
  - Then, the phone and tower (gNB) do the same for radio-level messages (AS security).
    i.e.AS (Access Stratum) Security Mode Command and Complete: The gNB and UE establish AS level security.

- **Subscriber info retrieval**: AMF gets user details from UDM and marks itself as the phone's current handler.

  i.e. **UDM Registration**: The AMF interacts with the UDM to retrieve subscriber data, and registers itself with the UDM as the serving AMF for the UE.
- **Policy setup**: AMF coordinates with PCF to define rules for how the phone can use the network (e.g., data limits, QoS, session rules).

  i.e. **PCF (Policy Control Function) Interaction**: The AMF creates a policy association with the PCF for policy control during the session.

3. PDU session establishment

- **Session planning**: AMF talks to SMF to plan how the phone will send/receive data. SMF picks a UPF to handle the actual data flow. i.e. **SMF context setup**: The new AMF updates the SMF (Session Management Function) context, and the SMF selects an appropriate UPF (User Plane Function) for the PDU session.
- **UPF preparation**: SMF tells UPF to get ready to send/receive data using PFCP messages.

  i.e. **UPF Configuration**: The SMF sends a PFCP (Packet Flow Control Protocol) Session Modification Request to the UPF to prepare it to receive uplink data and to buffer downlink data.
- **Final setup messages:** AMF sends a “Registration Accept” to the phone. gNB sends “RRC Reconfiguration” to set up the data channel (DRB).

  i.e. **Final setup messages** as Registration Accept and RRC Reconfiguration: The AMF sends a Registration Accept message to the UE via the gNB. The gNB then sends an RRC Reconfiguration message to the UE, establishing the default PDU session and Data Radio Bearer (DRB) for data transfer.

- **Confirmation**: Phone replies with "Registration Complete."
  i.e. **Registration Complete**: The UE sends a Registration Complete message to the AMF, indicating the successful completion of the registration procedure.
- **Data flow begins**: SMF finalizes the setup with UPF, and the phone or UE can now send and receive data.
  i.e. **Final SMF-UPF/PFCP Update**: The SMF updates the PFCP session with the UPF to enable downlink data transmission, and uplink/downlink data flow begins.

Above summarizes the key steps involved in a 5G SA Call Flow, focusing on the registration and initial PDU session establishment. Note that this is a simplified view and the actual message exchanges are more detailed.

Working with Wireshark

Wireshark is a **free, open-source network protocol analyser** used to capture and inspect data packets in real time. Think of it as a microscope for network traffic—revealing the inner workings of communication between devices.

- Originally named **Ethereal**, it was developed by Gerald Combs in 1997 and renamed Wireshark in 2006.
- It's widely used by **network engineers, security professionals, developers**, and even educators to understand protocol behaviour and troubleshoot issues.

Key Features

- **Live Packet Capture**: From Ethernet, WLAN, Bluetooth, USB, and more.
- **Protocol Support**: Over 2,000 protocols including TCP/IP, HTTP, DNS, SIP, etc.
- **Filtering & Search**: Apply display filters to isolate specific traffic.
- **Colour Coding**: Visual cues for different packet types.
- **Export & Import**: Supports multiple capture formats like .pcap, .pcapng.

Getting Started

Installation

- **Windows**: Download from Wireshark Downloads
- **Linux (Ubuntu)**:

Basic Workflow

1. Select a network interface.
2. Start capturing packets.
3. Apply filters (e.g., http, ip.addr == 192.168.1.1).
4. Inspect packet details in the decode pane.
5. Save or export the capture for documentation or further analysis.

Step-by-Step Workflow in Wireshark

1. Install Wireshark

- **Windows/macOS**: Download from Wireshark Downloads
- **Linux (Ubuntu)**:

```
sudo add-apt-repository ppa:wireshark-dev/stable
sudo apt-get update
sudo apt-get install wireshark
sudo wireshark
```

2. Launch the Application

- Open Wireshark and grant necessary permissions for packet capture.
- You'll see a list of available **network interfaces** (e.g., Ethernet, Wi-Fi).

3. Start Capturing Packets

- Select the desired interface.
- Click the **green shark fin icon** or press Ctrl + E to begin capture.
- Packets will start appearing in real time.

4. Apply Display Filters

Use filters to narrow down the traffic:

- http – Show only HTTP traffic
- ip.addr == 192.168.1.1 – Filter by IP address
- tcp.port == 443 – Filter by port

You can combine filters using logical operators:

Plaintext (ip.src == 192.168.1.1) && (tcp.port == 80)

5. Inspect Packets

Click on any packet to view:

- **Summary pane**: Time, source, destination, protocol, length
- **Decode pane**: Layer-by-layer breakdown (Ethernet, IP, TCP, etc.)
- **Hex pane**: Raw data in hexadecimal and ASCII

6. Stop and Save Capture

- Click the **red square icon** or press Ctrl + E again to stop.
- Go to File > Save As to store the capture in .pcap or .pcapng format.

7. Analyse and Export

- Use built-in statistics: Statistics > Protocol Hierarchy, IO Graphs, etc.
- Export filtered packets or summaries for reporting.

8. Use Capture Files

- Open .pcap files from previous sessions via File > Open.

Working with MobaXterm

MobaXterm is a **Windows-based terminal emulator** that integrates:

- **SSH, RDP, VNC, FTP, SFTP, X11** protocols
- A built-in **X11 server** for graphical Linux apps
- A **Unix-like shell** with tools like bash, ls, grep, etc.

It's ideal for developers, sysadmins, and network engineers who need remote access to Linux/Unix servers from Windows.

Step-by-Step: Using MobaXterm

1. Download & Install

- Visit MobaXterm Downloads
- Choose **Home Edition (Portable or Installer)**
- Run the executable—no installation needed for the portable version

2. Start a New Session

- Click **"Session"** in the top-left corner
- Choose **SSH** for remote Linux access
- Enter:
  - **Remote Host** (IP or domain)
  - **Username** (e.g., ec2-user, ubuntu)
  - **Port** (default SSH is 22)
  - Optionally, attach your **.pem private key** for authentication

3. Enable X11 Forwarding (Optional)

- Check the box for **X11-Forwarding** to run graphical apps remotely
- MobaXterm's built-in X server handles GUI rendering

4. Connect and Work

- Click **OK** to launch the session
- You'll see a terminal window and a file browser (left pane)

- You can:
  - Run shell commands
  - Transfer files via drag-and-drop
  - Monitor server resources using **Remote Monitoring**

5. Save and Secure Sessions

- Save sessions for quick access later
- Set a **master password** to encrypt stored credentials

5G Lab Book

Part -1: 5G Features

Data Rate support in a 5G network

This experiment aims to verify end to end data rate (uplink (40 mbps and downlink 340 mbps) of 5G network utilizing IPERF tool installed in NG Core and UE.

Key Components of 5G Data Rate Testing

Test Setup

- **Network Configuration**: Include both Standalone (SA) and Non-Standalone (NSA) modes. gNB should be configured as per RFP & it should be connected with 5G Next Generation Core (NGC). UE should be registered with NGC with data working.
- **Device Selection**: Use real 5G-capable devices (UE)
- **Frequency Bands**: Test across Sub-6 GHz and mmWave bands.
- **Virtualization Tools**: Incorporate Network Function Virtualization (NFV) and Cloud-Native Network Function (CNF) based setups for cloud-native testing.

Testing methodology:

- Use iperf tool to verify the data rate between UE and 5G NGC.

Install IPERF tool on 5G NGC and UE and attach the UE.

Downlink Test:

On the UE side: Open the iperf application in UE,

Run iperf –s –u –i1

On the NGC side: Open the terminal in NGC,

Run iperf –c <UE ip> –u –i1 –b200M –t 60

Uplink Test:

On the UE side: Open the iperf application in UE,

Run iperf –c [UE ip] –u –i1 –b200M –t 60

On the NGC side: Open the terminal in NGC,

Run iperf –s –u –i1

- Use MobaXterm, the terminal emulator for Windows to connect gNB, to verify the DL & UL throughput in gNB
  Open MobaXterm in laptop, to initiate the ssh connection to the gNB
  Type root@ [gNB ip]
  Run, “watch cat gnb_du_cell_stats.txt”

```
Timestamp: 2025-08-06 22:55:12

Cell ID    | PCI | Band | DL BW | UL BW | RSRP (dBm) | SINR (dB) | Active UEs | DL Throughput (Mbps) | UL Throughput (Mbps)
----------------------------------------------------------------------------------------------------------
1001       | 10  | n78  | 100MHz| 100MHz| -85        | 20        | 1          | 450.5                | 120.3
1002       | 11  | n78  | 100MHz| 100MHz| -88        | 18        | 1          | 390.2                | 98.7
1003       | 12  | n28  | 20MHz | 20MHz | -92        | 12        | 1          | 120.4                | 45.6
```

Acceptance Criteria

Verify that the data transfer is successful between the Server and the Client & vice versa.

Result: DL Data Rate: >340Mbps UL Data Rate: >40Mbps

Possible Test Scenarios

- **Peak Data Rate Measurement**: Evaluate maximum achievable throughput under ideal conditions.
- **Real-World Usage Simulation**: Include mobility, handovers, and network congestion.
- **Application-Level Testing**: Stream video, download files, and run latency-sensitive apps.
- **QoS Validation**: Ensure service differentiation across slices and applications.

Testing Methodologies

- **Drive Testing**: Measure data rates in different geographic areas using mobile test units.
- **Lab Testing**: Controlled environment testing for protocol and performance validation.
- **End-to-End Testing**: From user equipment (UE) to core network, including transport layers.
- **Automation & Scripting**: Use tools like TEMS, Viavi, or Keysight for repeatable test cases.
- **Lifecycle Monitoring**: Continuous evaluation during deployment and operation phases.

latency in a 5G network

To test **latency in a 5G network**, you'll need a procedure that captures **round-trip time (RTT)** between endpoints—typically between a **User Equipment (UE)** and the **5G Core (NGC)** or an external server. Here's a structured approach tailored for field or lab testing:

Test Setup Overview

Component	**Role**
UE (e.g., smartphone or CPE)	Acts as the client
gNB	Connects UE to the 5G network
NGC	Core network, acts as server
Tool	ping, iperf, or latency probes

Latency Test Procedure

Option 1: Using ping (ICMP-based latency)

1. **On UE terminal:**

```
ping [NGC IP] -i 0.2 -c 100
```

- -i 0.2: Sends a ping every 200 ms
- -c 100: Sends 100 packets
- Replace [NGC IP] with the actual IP of the core or test server

2. **Observe Output:**

txt

```
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=9.23 ms

...

--- 192.168.1.1 ping statistics ---

100 packets transmitted, 100 received, 0% packet loss

rtt min/avg/max/mdev = 8.91/9.15/9.42/0.12 ms
```

Option 2: Using iperf for UDP latency

1. **On NGC (Server):**

```
iperf -s -u
```

2. **On UE (Client):**

```
iperf -c [NGC IP] -u -b10M -t30 --latency
```

   - --latency: Enables latency measurement (if supported by version)
   - -b10M: Sends at 10 Mbps
   - -t30: Runs for 30 seconds

3. **Output Sample:**

```
[ ID] Interval      Transfer     Bandwidth      Jitter   Latency   Lost/Total
[  3] 0.0-1.0 sec    1.25 MB      10 Mbps       0.12 ms  8.5 ms    0/1000 (0%)
```

Best Practices

- **Test at different times** to capture network load impact.
- **Use multiple UEs** to simulate real-world traffic.
- **Log results** for comparison across locations or configurations.
- **Correlate with throughput and jitter** for full QoS analysis.

Channel BW and Radio Band Support in 5G network

Test and find Channel BW and Radio Band Support in 5G network

This is to show Channel BW configuration with different values (20 MHz, 40 MHz, …, 100 MHz) in the configuration file and thereafter, observe changes in the Log file. Radio Band Support is found in the gNB configuration file.

Set Steps

End to End systems to be configured with valid network parameters and the gNB is brought up and attach UE with gNB. Configure different bandwidths 20MHz, 40MHz, 60MHz, 80MHz, 100MHz in gNB

Procedure:

gNB Config File:

```
<bSChannelBwDL>100MHZ</bSChannelBwDL><dlEarfcn>623333</dlEarfcn><nrFreqBand>78</nrFreqBand><subCarrierCfg><offsetToCarrier>0</offsetToCarrier><carrierBw>273</carrierBw><subCarrierSpacing>KHz30</subCarrierSpacing>
```

UE Logs:

```
servingCellConfigCommon
{
downlinkConfigCommon
{
frequencyInfoDL
{
frequencyBandList
{
{
freqBandIndicatorNR 78
}
},
offsetToPointA 0, scs-SpecificCarrierList
{
{
offsetToCarrier 0, subcarrierSpacing kHz30, carrierBandwidth 273
```

100MHZ bandwidth verified in Netmonster android application

Data rate at different Channel BW in 5G network

Test and observe data rate at different Channel BW in 5G network

Configure gNB with different BW

Login to gNB -> Navigate to config directory -> edit Config File -> Start the gNB service

```
cd /etc/gnb/          # Navigate to config directory

nano gnb.conf         # Edit the configuration file

./gnb_start.sh        # Start the gNB service (example script)
```

Editing the Configuration file

```
# gNB Configuration File

[GENERAL]
nr_band = n78
duplex_mode = TDD
channel_bandwidth = 20MHz                    // Put different values like 20, 40, 60 etc
subcarrier_spacing = 30kHz
center_frequency = 3600000000  # in Hz
nr_arfcn = 620000
pci = 10
tac = 1001

[PLMN]
mcc = 001
mnc = 01

[INTERFACE]
gnb_ip = 192.168.1.10
amf_ip = 192.168.1.100
ngap_port = 38412

[SCHEDULING]
slot_allocation = dynamic
max_mimo_layers = 4
```

For each BW configured, repeat the experiment > To test and verify data rate support in a 5G network > Tabulate for observation of the data rate variation with Channel BW.

Explanation of how **data throughput varies with channel bandwidth in 5G**, based on current standards and practical deployments:

Relationship Between Channel Bandwidth and Data Throughput in 5G

Fundamental Concept

- **Channel Bandwidth**: The range of frequencies allocated for data transmission.
- **Data Throughput**: The actual rate at which data is transmitted over the network.
- In 5G, wider bandwidth = more resource blocks (RBs) = higher data capacity.

Variation of Data Throughput with Bandwidth (FR1, SCS = 30 kHz)

Channel Bandwidth (MHz)	Max RBs	Transmission Bandwidth Proportion	Approx. Data Rate Potential
5	11	79.2%	Low (~100 Mbps)
10	24	86.4%	Moderate (~200–300 Mbps)
20	51	91.8%	High (~500–700 Mbps)
50	133	95.8%	Very High (~1–2 Gbps)
100	273	98.3%	Peak (~5 Gbps or more)

Key Insights

- **Higher Bandwidth = More RBs**: Each RB carries data; more RBs mean more data per unit time.
- **Efficiency Improves with Bandwidth**: Transmission bandwidth proportion increases with channel bandwidth, meaning less spectrum is wasted on guard bands.
- **SCS Matters**: Subcarrier spacing (e.g., 15 kHz, 30 kHz, 60 kHz) affects how many RBs fit into a given bandwidth.

Practical Considerations

- **UE Capability**: Not all user equipment supports wide bandwidths.

- **Deployment Band**: Low-band (e.g., n28) may use 20 MHz, while mmWave (e.g., n258) can go up to 400 MHz.
- **Network Load & Scheduling**: Real-world throughput depends on user density, scheduling algorithms, and MIMO layers.

Max. number of UE and Observation of simultaneous active devices

Setting up max number of UE and Observation of simultaneous active devices

This experiment would use MobaXterm and Wireshark for verifying max number of PDU session. In MobaXterm, Run, “watch cat gnb_du_cell_stats.txt” Command to check no of UE & Radio Network Temporary Identifier (RNTI) values. In Wireshark, apply nas-5gs filter to verify max no. of PDU session after opening ‘pcap’ log file

SetUp

End to End systems to be configured with valid network parameters and the gNB is brought up.

Procedure

Configure MaxBandwidth, PRB, SCS, OFDM symbols DuplexMode, Band, maxNumUEs in gNB & maxNumUEs, maxPduSessions in NGC

Login to gNB -> configure gNB

To Login gNB via MobaXterm. Open mobaxtrem in laptop,
Type, root@ [gNB IP]
Run, tcpdump -i [Ethernet/optical interface of gNB] -w filename. pcap

```
./gnb --nr-band n78 \
    --duplex-mode TDD \
    --channel-bandwidth 20 \
    --subcarrier-spacing 30 \
    --max-prbs 51 \
    --ofdm-symbols 14 \
    --max-num-ues 32 \
    --center-frequency 3600000000
# Launch NGC (e.g., AMF/SMF) with UE and session limits
./ngc --max-num-ues 32 \
    --max-pdu-sessions 64
```

Max No of UE should successfully register with NGC

Open Wireshark app in laptop.
Open Wireshark >click File > open pcap log (filename. pcap) >
Apply nas-5gs filter to verify max no. of PDU session

Result: Verify up to 64 nos of active PDU session.

To verify Max no of active UE in gNB,
Open mobaxtrem in laptop
Type, root@ [gNB IP]
Run, “watch catgnb_du_cell_stats.txt” Command to check no of UE &RNTI values

Result: Verify up to 64 no’s of Active Users.

Network Slicing

Find configuration and parameters for Network Slicing

Network Slicing parameters is configured in gnb_config.xml of gNB.

Setup

End to End systems are configured & gNB is radiating. UE should successfully register with NGC.

Procedure

To **verify supported NSSAI** in the **NG Setup Response** for identifying available **network slices** in a 5G system, inspect the Supported S-NSSAI field within the NGAP (NG Application Protocol) signalling between the **gNB** and **AMF**.

Carry out verification of supported NSSAI in NGsetup Response for Network slicing.

Verify supported NSSAI inNGsetup ResponseNGSetupResponseprotocolIEs: 4 items
Item 0: id-AMFName
Item 1: id-ServedGUAMIList
Item 2: id-RelativeAMFCapacity
Item 3: id-PLMNSupportListProtocol
IE-Field id: id-PLMNSupportList (80)
criticality: reject (0) value

PLMNSupportList: 1 item
Item 0: PLMNSupportItem
pLMNIdentity: 00f110
Mobile Country Code (MCC): Unknown (001)
Mobile Network Code (MNC): Unknown (01)

```
sliceSupportList: 4 items
Item 0: SliceSupportItems-NSSAI
sST: 01
Item 1: SliceSupportItems-NSSAI
sST: 02
Item 2: SliceSupportItems-NSSAI
sST: 03
Item 3: SliceSupportItems-NSSAI
sST: 03

sD: 030609
```

Verify the configured Slice value in EMS Slice (NSSAI): SST: X, SD: XXX

VoNR support in the 5G network

Find and verify support for VoNR in the 5G network

This experiment is to verify that the 5G network supports VoNR.

First verify that UE registration is successful and QoS Flow Identifier (QFI-5) is established via ‘pcap’ log. Initiate VoNR call between two UEs. The 5G NR Radio creates dedicated bearer with QFI-1. It should be verified that VoNR call is successful & QFI-1 is established in pcap log. With Wireshark, open ‘pcap’ log file to apply filter for GTP-U, rtp, sip, UDP.

Set up

gNB should be configured as per RFP & its connected with NGC with Sngrep IMS node.
Perform UE registration. The UE shall also register with Sngrep IMS node.
Initiate VoNR call. The 5G NR Radio shall create dedicated bearer withQFI-1.

Procedure

To verify gNB is configured as per RFP& its connected with NGC with Sngrep IMS node.

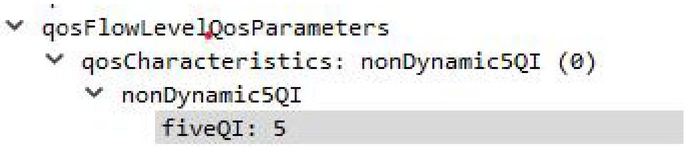
Testing methodology:
Verify that UE registration successful &QFI-5 is established for IMS via pcap log,

Steps:
To Login gNB via mobaxtrem, Open mobaxtrem in laptop, Type, root@ [gNB IP]PW: root
Run, tcp dump -i[Ethernet/optical interface of gNB] -w [filename].pcap

Validation:
Open Wireshark app in laptop. Open Wireshark >click File > open pcap log (filename. pcap)>
Apply nas-5gs || sip|| gtp
Filter in pcap log.

Result:

QFI-5 is established.



```
qosFlowLevelQosParameters
  qosCharacteristics: nonDynamic5QI (0)
    nonDynamic5QI
      fiveQI: 5
```

Verify that VoNR call is successful &QFI-1 is established in pcap log.qos_rule_precedence = 255(0xff) qfi_present = 1 (0x1)segregation = 0 (0x0)qfi = 1(0x1)

Open Wireshark >click File > open pcap log(filename. pcap) >
Apply gtp || gtpu, rtp, sip,
UDP filters in pcap log

Result:

OFI-1 Is established.
Verified gNB supports VoNR via Wireshark logs.

Communication Protocol deployed in the 5G Network

Find Protocol deployed in the 5G Network for communication

This experiment involves UE registration with APN protocol IPv4, IPv6 and IPv4/IPv6 configuration then validation using Wireshark. Filter nas-5gs in 'pcap' log file to find Transport Layer Address as IPv4, IPv6 and IPv4/IPv6 addresses respectively.

Setup

End to End systems to be configured with valid network parameters and the gNB is brought up and attach UE with gNB.
Perform UE registration with APN protocolIPv4/IPv6configuration.

Procedure

To verify End to End systems are configured & gNB is radiating.

Testing methodology:

To verify that IPv4/IPv6 in PDU in Wireshark log,

Steps:

To Login gNB via mobaxtrem,
Open mobaxtrem in laptop,
Type, root@ [gNB IP]
Run, tcpdump -i [Ethernet/optical
interface of gNB] -w filename.pcap

Validation:
Open Wireshark app in laptop.
Open Wireshark > File > open pcap log > Apply nas-5gs> PDU session establishment accept>IPv4/IPv6
Transport Layer Address (IPv4):
192.168.15.10 session

Result:

IPv4:192.168.11.9
IPv6:2001:4490:c1c:f8b1:25a7:eb80:2bfb:fca5

Security Protocol in the 5G network

Find the deployed Security Protocol in the 5G network

This is to find the security feature available in the 5G Network. Steps involved are configuring IPSec in gNB with NGC, attaching UE with gNB and then validating with Wireshark by filtering ip addresses in 'pcap' log file. This will show that IPSec is working as well as Control plan and User plan are encrypted.

Set up

End to End systems to be configured with valid network parameters and the gNB is brought up and attach UE with gNB.
Configure gNB with Network slicing.
Configure Ipsec in gNB with NGC

Procedure

To verify End to End systems are configured &gNB is radiating.

Testing methodology:

To verify that each slice security policies & data encryption,

Steps:

To Login gNB via mobaxtrem, Open mobaxtrem in laptop,
Type, root@ [gNB IP] PW: root
Run, tcp dump -i [Ethernet/optical interface of gNB] -w filename.pcap

Validation:

Open Wireshark app in laptop.
Open Wireshark > File > open
pcap log > Apply filter,
ip. addr == <Slice_1_IP_address>
|| ip. addr == <Slice_2_IP_address> tls || ssl || ipsec

Result: Control plane & User plane packets are encrypted with IPsec Protocol

TDD Duplexing Mode

To verify gNB supports TDD Duplexing Mode

The gNB must be configured with NR-TDD parameters that define how uplink (UL) and downlink (DL) transmissions are scheduled over time:

Frame Structure: Typically 1 ms duration, divided into slots.

Slot Formats: Type 0 (symmetric), Type 1 (asymmetric), Type 2 (short transmissions).

DL/UL Allocation: Flexible slot assignment based on traffic needs.

Special Subframes: Used for synchronization and control signalling.

Subframe Patterns: Common patterns include 3:1, 2:2, 2:3, etc.

Dynamic Reconfiguration: gNB can adapt slot allocation based on traffic load.

Synchronization & Interference Management: Critical for TDD operation due to shared frequency usage.

Set up

Integrated 5G NR Radio should be configured with TDD duplexing mode& it's connected with 5G NGC.
gnb = nrGNB('DuplexMode','TDD');
Configured Uplink & Downlink frequencies (should be same) in gnb_config.xml of gNB.

UE should successfully register with NGC.

Procedure

Open the "Network Signal Guru" application in Android UE and see the “Mode” in which the UE is registered

Result:

Mode: TDD
To verify the configuration in gNB Open mobaxtrem in laptop,
Type, root@ [gNB IP] PW:
Type path: /du/config/
Open gnb_config.xml file then check the Mode type

Result:

In a live network:
Check gNB configuration files or logs for TDD slot formats.
Use tools like Wireshark or iPerf to observe traffic patterns.
Look for DL/UL scheduling on the same frequency but alternating time slots.

MIMO configuration in gNB

To find and verify MIMO configuration in gNB

2T2R is a basic MIMO setup, often used in rural or low-traffic areas.

For advanced deployments, gNBs may support 4T4R, 8T8R, or even Massive MIMO (64T64R).

Ensure the UE also supports 2R to utilize full 2T2R benefits.

Set up

End to End systems to be configured with valid network parameters and gNB is brought up and attach UE with gNB.

Configure 2T2R MIMO configuration in gnb_config.xml of gNB.

Procedure

Inspect gNB Configuration

Use the Element Management System (EMS) or Network Management System (NMS).

Navigate to the radio unit settings and confirm:

Number of Tx/Rx ports

Supported MIMO layers

Active antenna configuration

RRC Signalling Analysis

Capture UE Capability Information and RRC Connection Setup Complete messages using tools like Wireshark.

Look for MIMO-related IE (Information Elements) such as:

supportedMIMO-Layers

antennaPortsCount

transmissionMode

Use Diagnostic Tools
Run diagnostics using iPerf, RantCell, or vendor-specific CLI tools.

Measure throughput and verify if dual streams are active (indicative of 2T2R).

To verify End to End systems are configured & gNB is radiating.

Testing methodology:

Verify gNB Config file <dlNumAntPorts>2</dlNum AntPorts>
<ulNumOfAntPorts>2</ulN
umOfAntPorts>

Verify at RRC Reconfiguration message in pcab file pdsch-ServingCellConfig setup:
{
nrofHARQProcessesForPDSCH n16, **maxMIMO-Layers 2**

5G Lab Book

Part -2: 5G NG Core

Protocol Deployment on NG Core Server

In 5G, protocol deployment involves installing and configuring core network functions.

All required protocols are bundled into a single ISO image for simplified deployment.

Control Plane Protocols: Responsible for signaling and control information.

- Non-Access Stratum (NAS)
- Next Generation Application Protocol (NGAP)
- Stream Control Transmission Protocol (SCTP)
- Packet Forward Control Protocol (PFCP)

User Plane Protocols: Handle the actual user data transmission.

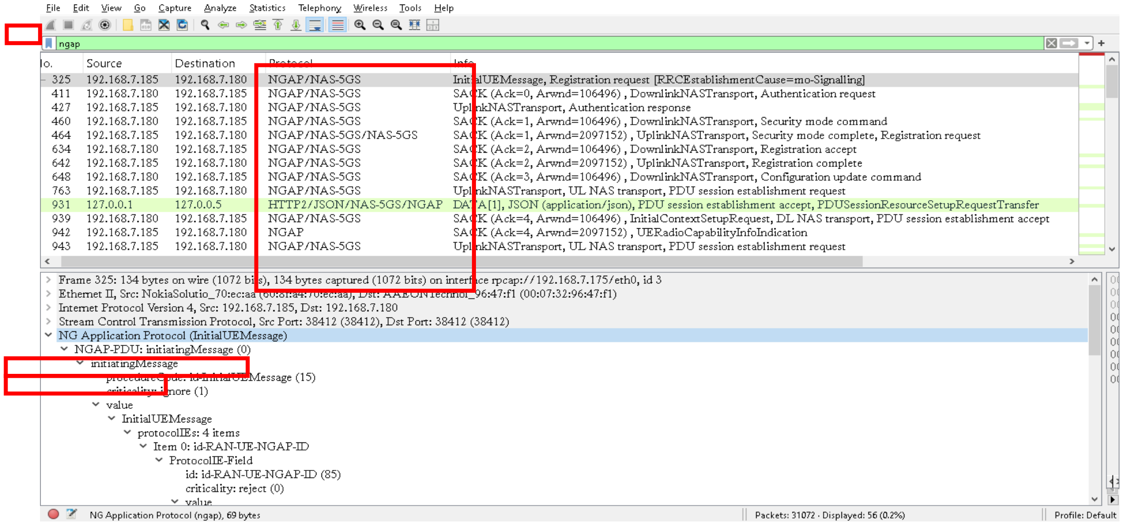
GPRS Tunnelling Protocol – User Plane (GTP-U)

5G Service Protocols: Manage NF communication and session policies.

Hyper Text Transfer Protocol /2 (HTTP/2)

NAS (Non-Access Stratum)

- It handles signaling between the User Equipment (UE) and the 5G Core Network (AMF).
- NAS operates independently of the RAN (Radio Access Network) and is responsible for UE authentication, mobility management, and session management.



File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ngap

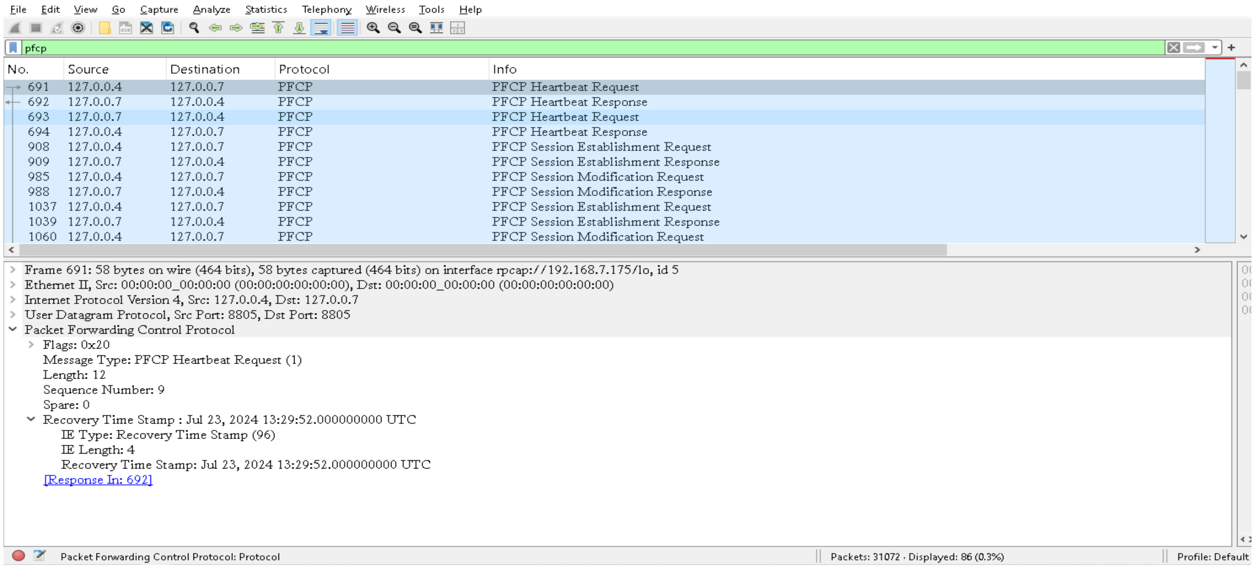
lo.	Source	Destination	Protocol	Info
325	192.168.7.185	192.168.7.180	NGAP/NAS-5GS	InitialUEMessage, Registration request [RRCEstablishmentCause=mo-Signalling]
411	192.168.7.180	192.168.7.185	NGAP/NAS-5GS	SACK (Ack=0, Arwnd=106496) , DownlinkNASTransport, Authentication request
427	192.168.7.185	192.168.7.180	NGAP/NAS-5GS	UplinkNASTransport, Authentication response
460	192.168.7.180	192.168.7.185	NGAP/NAS-5GS	SACK (Ack=1, Arwnd=106496) , DownlinkNASTransport, Security mode command
464	192.168.7.185	192.168.7.180	NGAP/NAS-5GS/NAS-5GS	SACK (Ack=1, Arwnd=2097152) , UplinkNASTransport, Security mode complete, Registration request
634	192.168.7.180	192.168.7.185	NGAP/NAS-5GS	SACK (Ack=2, Arwnd=106496) , DownlinkNASTransport, Registration accept
642	192.168.7.185	192.168.7.180	NGAP/NAS-5GS	SACK (Ack=2, Arwnd=2097152) , UplinkNASTransport, Registration complete
648	192.168.7.180	192.168.7.185	NGAP/NAS-5GS	SACK (Ack=3, Arwnd=106496) , DownlinkNASTransport, Configuration update command
763	192.168.7.185	192.168.7.180	NGAP/NAS-5GS	UplinkNASTransport, UL NAS transport, PDU session establishment request
931	127.0.0.1	127.0.0.5	HTTP2/JSON/NAS-5GS/NGAP	DATA[1], JSON (application/json), PDU session establishment accept, PDUSessionResourceSetupRequestTransfer
939	192.168.7.180	192.168.7.185	NGAP/NAS-5GS	SACK (Ack=4, Arwnd=106496) , InitialContextSetupRequest, DL NAS transport, PDU session establishment accept
942	192.168.7.185	192.168.7.180	NGAP	SACK (Ack=4, Arwnd=2097152) , UERadioCapabilityInfoIndication
943	192.168.7.185	192.168.7.180	NGAP/NAS-5GS	UplinkNASTransport, UL NAS transport, PDU session establishment request

```
> Frame 325: 134 bytes on wire (1072 bits), 134 bytes captured (1072 bits) on interface rpcap://192.168.7.175/eth0, id 3
> Ethernet II, Src: NokiaSolutio_70:ec:aa (60:8f:a4:70:ec:aa), Dst: AAEONTechnol_96:47:f1 (00:07:32:96:47:f1)
> Internet Protocol Version 4, Src: 192.168.7.185, Dst: 192.168.7.180
> Stream Control Transmission Protocol, Src Port: 38412 (38412), Dst Port: 38412 (38412)
v NG Application Protocol (InitialUEMessage)
  v NGAP-PDU: initiatingMessage (0)
    v initiatingMessage
        procedureCode: id-InitialUEMessage (15)
        criticality: ignore (1)
      v value
        v InitialUEMessage
          v protocolIEs: 4 items
            v Item 0: id-RAN-UE-NGAP-ID
              v ProtocolIE-Field
                  id: id-RAN-UE-NGAP-ID (85)
                  criticality: reject (0)
                v value
```

NG Application Protocol (ngap), 69 bytes | Packets: 31072 · Displayed: 56 (0.2%) | Profile: Default

PFCP (Packet Forwarding Control Protocol)

- Communication between SMF and UPF.
- Controls the establishment, modification, and deletion of PDU sessions.
- Manages QoS policies and traffic rules.



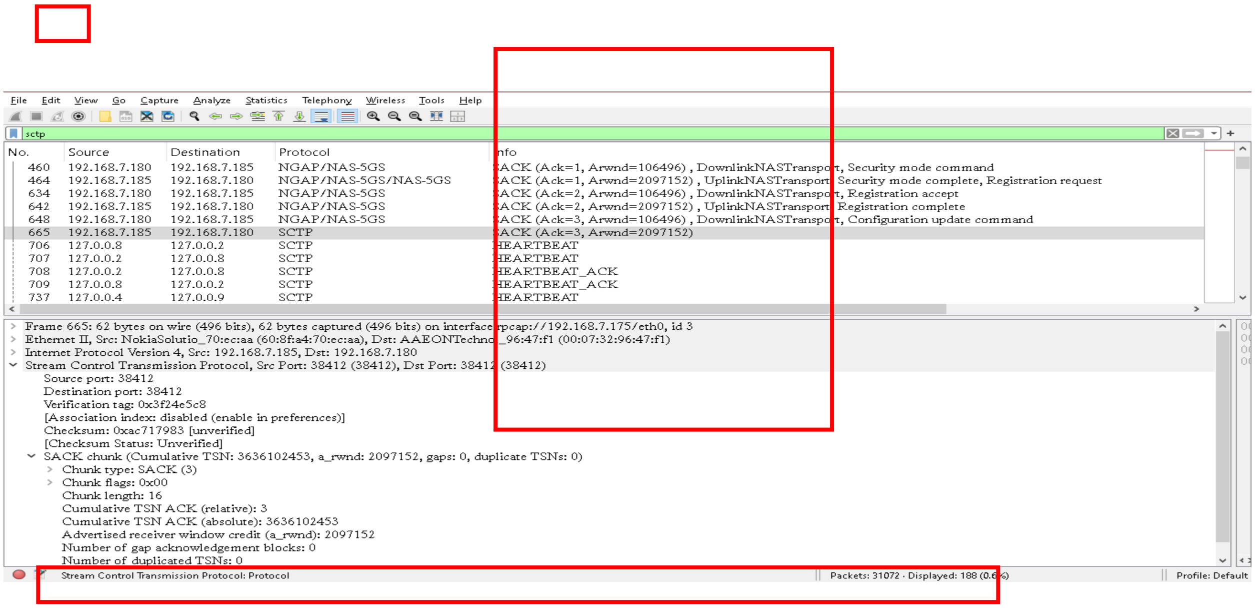
No.	Source	Destination	Protocol	Info
691	127.0.0.4	127.0.0.7	PFCP	PFCP Heartbeat Request
692	127.0.0.7	127.0.0.4	PFCP	PFCP Heartbeat Response
693	127.0.0.7	127.0.0.4	PFCP	PFCP Heartbeat Request
694	127.0.0.4	127.0.0.7	PFCP	PFCP Heartbeat Response
908	127.0.0.4	127.0.0.7	PFCP	PFCP Session Establishment Request
909	127.0.0.7	127.0.0.4	PFCP	PFCP Session Establishment Response
985	127.0.0.4	127.0.0.7	PFCP	PFCP Session Modification Request
988	127.0.0.7	127.0.0.4	PFCP	PFCP Session Modification Response
1037	127.0.0.4	127.0.0.7	PFCP	PFCP Session Establishment Request
1039	127.0.0.7	127.0.0.4	PFCP	PFCP Session Establishment Response
1060	127.0.0.4	127.0.0.7	PFCP	PFCP Session Modification Request

```
> Frame 691: 58 bytes on wire (464 bits), 58 bytes captured (464 bits) on interface rpcap://192.168.7.175/lo, id 5
> Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
> Internet Protocol Version 4, Src: 127.0.0.4, Dst: 127.0.0.7
> User Datagram Protocol, Src Port: 8805, Dst Port: 8805
v Packet Forwarding Control Protocol
    > Flags: 0x20
      Message Type: PFCP Heartbeat Request (1)
      Length: 12
      Sequence Number: 9
      Spare: 0
    v Recovery Time Stamp : Jul 23, 2024 13:29:52.000000000 UTC
          IE Type: Recovery Time Stamp (96)
          IE Length: 4
          Recovery Time Stamp: Jul 23, 2024 13:29:52.000000000 UTC
      [Response In: 692]
```

Packet Forwarding Control Protocol: Protocol | Packets: 31072 · Displayed: 86 (0.3%) | Profile: Default

SCTP (Stream Control Transmission Protocol)

- Transport layer protocol for NGAP communication.
- Provides reliable message delivery between gNB and AMF.
- Supports multi-streaming to prevent head-of-line blocking.

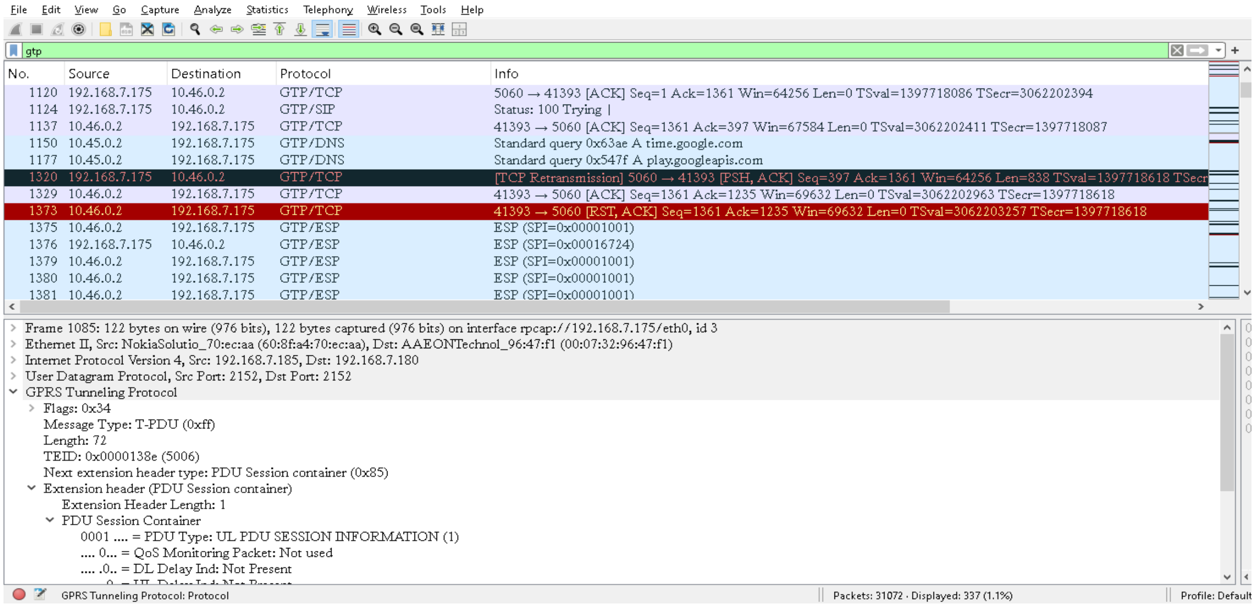


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
sctp

No.   Source          Destination     Protocol              Info
 460  192.168.7.180   192.168.7.185   NGAP/NAS-5GS          SACK (Ack=1, Arwnd=106496) , DownlinkNASTransport, Security mode command
 464  192.168.7.185   192.168.7.180   NGAP/NAS-5GS/NAS-5GS  SACK (Ack=1, Arwnd=2097152) , UplinkNASTransport, Security mode complete, Registration request
 634  192.168.7.180   192.168.7.185   NGAP/NAS-5GS          SACK (Ack=2, Arwnd=106496) , DownlinkNASTransport, Registration accept
 642  192.168.7.185   192.168.7.180   NGAP/NAS-5GS          SACK (Ack=2, Arwnd=2097152) , UplinkNASTransport, Registration complete
 648  192.168.7.180   192.168.7.185   NGAP/NAS-5GS          SACK (Ack=3, Arwnd=106496) , DownlinkNASTransport, Configuration update command
 665  192.168.7.185   192.168.7.180   SCTP                  SACK (Ack=3, Arwnd=2097152)
 706  127.0.0.8       127.0.0.2       SCTP                  HEARTBEAT
 707  127.0.0.2       127.0.0.8       SCTP                  HEARTBEAT
 708  127.0.0.2       127.0.0.8       SCTP                  HEARTBEAT_ACK
 709  127.0.0.8       127.0.0.2       SCTP                  HEARTBEAT_ACK
 737  127.0.0.4       127.0.0.9       SCTP                  HEARTBEAT

> Frame 665: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface rpcap://192.168.7.175/eth0, id 3
> Ethernet II, Src: NokiaSolutio_70:ec:aa (60:8f:a4:70:ec:aa), Dst: AAEONTechno_96:47:f1 (00:07:32:96:47:f1)
> Internet Protocol Version 4, Src: 192.168.7.185, Dst: 192.168.7.180
v Stream Control Transmission Protocol, Src Port: 38412 (38412), Dst Port: 38412 (38412)
    Source port: 38412
    Destination port: 38412
    Verification tag: 0x3f24e5c8
    [Association index: disabled (enable in preferences)]
    Checksum: 0xac717983 [unverified]
    [Checksum Status: Unverified]
  v SACK chunk (Cumulative TSN: 3636102453, a_rwnd: 2097152, gaps: 0, duplicate TSNs: 0)
    > Chunk type: SACK (3)
    > Chunk flags: 0x00
      Chunk length: 16
      Cumulative TSN ACK (relative): 3
      Cumulative TSN ACK (absolute): 3636102453
      Advertised receiver window credit (a_rwnd): 2097152
      Number of gap acknowledgement blocks: 0
      Number of duplicated TSNs: 0

Stream Control Transmission Protocol: Protocol  ||  Packets: 31072 · Displayed: 188 (0.6%)  ||  Profile: Default
```

GTP-U (GPRS Tunneling Protocol – User Plane)

- Transfers user data between gNB and UPF.
- Encapsulates IP packets for tunneling over the 5G network.
- Carries both DL and UL traffic.



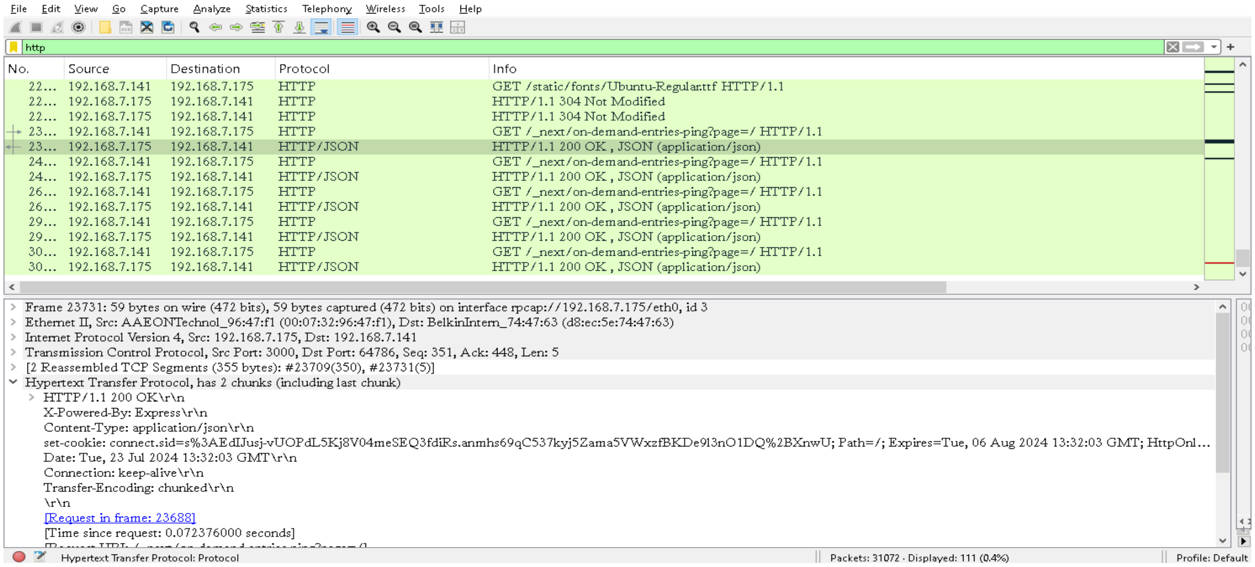
No.	Source	Destination	Protocol	Info
1120	192.168.7.175	10.46.0.2	GTP/TCP	5060 → 41393 [ACK] Seq=1 Ack=1361 Win=64256 Len=0 TSval=1397718086 TSecr=3062202394
1124	192.168.7.175	10.46.0.2	GTP/SIP	Status: 100 Trying |
1137	10.46.0.2	192.168.7.175	GTP/TCP	41393 → 5060 [ACK] Seq=1361 Ack=397 Win=67584 Len=0 TSval=3062202411 TSecr=1397718087
1150	10.45.0.2	192.168.7.175	GTP/DNS	Standard query 0x63ae A time.google.com
1177	10.45.0.2	192.168.7.175	GTP/DNS	Standard query 0x547f A play.googleapis.com
1320	192.168.7.175	10.46.0.2	GTP/TCP	[TCP Retransmission] 5060 → 41393 [PSH, ACK] Seq=397 Ack=1361 Win=64256 Len=838 TSval=1397718618 TSecr
1329	10.46.0.2	192.168.7.175	GTP/TCP	41393 → 5060 [ACK] Seq=1361 Ack=1235 Win=69632 Len=0 TSval=3062202963 TSecr=1397718618
1373	10.46.0.2	192.168.7.175	GTP/TCP	41393 → 5060 [RST, ACK] Seq=1361 Ack=1235 Win=69632 Len=0 TSval=3062203257 TSecr=1397718618
1375	10.46.0.2	192.168.7.175	GTP/ESP	ESP (SPI=0x00001001)
1376	192.168.7.175	10.46.0.2	GTP/ESP	ESP (SPI=0x00016724)
1379	10.46.0.2	192.168.7.175	GTP/ESP	ESP (SPI=0x00001001)
1380	10.46.0.2	192.168.7.175	GTP/ESP	ESP (SPI=0x00001001)
1381	10.46.0.2	192.168.7.175	GTP/ESP	ESP (SPI=0x00001001)

```
> Frame 1085: 122 bytes on wire (976 bits), 122 bytes captured (976 bits) on interface rpcap://192.168.7.175/eth0, id 3
> Ethernet II, Src: NokiaSolutio_70:ec:aa (60:8f:a4:70:ec:aa), Dst: AAEONTechnol_96:47:f1 (00:07:32:96:47:f1)
> Internet Protocol Version 4, Src: 192.168.7.185, Dst: 192.168.7.180
> User Datagram Protocol, Src Port: 2152, Dst Port: 2152
v GPRS Tunneling Protocol
  > Flags: 0x34
    Message Type: T-PDU (0xff)
    Length: 72
    TEID: 0x0000138e (5006)
    Next extension header type: PDU Session container (0x85)
  v Extension header (PDU Session container)
      Extension Header Length: 1
    v PDU Session Container
        0001 .... = PDU Type: UL PDU SESSION INFORMATION (1)
        .... 0... = QoS Monitoring Packet: Not used
        .... .0.. = DL Delay Ind: Not Present
        .... ..0. = UL Delay Ind: Not Present
```

GPRS Tunneling Protocol: Protocol | Packets: 31072 · Displayed: 337 (1.1%) | Profile: Default

Hypertext Transfer Protocol /2 (HTTP/2)

- Used for signaling between NFs in Service-Based Architecture.
- Supports efficient and multiplexed data communication.
- Enables faster session management.



http

```
No.     Source         Destination    Protocol     Info
  22... 192.168.7.141  192.168.7.175  HTTP         GET /static/fonts/Ubuntu-Regular.ttf HTTP/1.1
  22... 192.168.7.175  192.168.7.141  HTTP         HTTP/1.1 304 Not Modified
  22... 192.168.7.175  192.168.7.141  HTTP         HTTP/1.1 304 Not Modified
  23... 192.168.7.141  192.168.7.175  HTTP         GET /_next/on-demand-entries-ping?page=/ HTTP/1.1
  23... 192.168.7.175  192.168.7.141  HTTP/JSON    HTTP/1.1 200 OK , JSON (application/json)
  24... 192.168.7.141  192.168.7.175  HTTP         GET /_next/on-demand-entries-ping?page=/ HTTP/1.1
  24... 192.168.7.175  192.168.7.141  HTTP/JSON    HTTP/1.1 200 OK , JSON (application/json)
  26... 192.168.7.141  192.168.7.175  HTTP         GET /_next/on-demand-entries-ping?page=/ HTTP/1.1
  26... 192.168.7.175  192.168.7.141  HTTP/JSON    HTTP/1.1 200 OK , JSON (application/json)
  29... 192.168.7.141  192.168.7.175  HTTP         GET /_next/on-demand-entries-ping?page=/ HTTP/1.1
  29... 192.168.7.175  192.168.7.141  HTTP/JSON    HTTP/1.1 200 OK , JSON (application/json)
  30... 192.168.7.141  192.168.7.175  HTTP         GET /_next/on-demand-entries-ping?page=/ HTTP/1.1
  30... 192.168.7.175  192.168.7.141  HTTP/JSON    HTTP/1.1 200 OK , JSON (application/json)
```

```
> Frame 23731: 59 bytes on wire (472 bits), 59 bytes captured (472 bits) on interface rpcap://192.168.7.175/eth0, id 3
> Ethernet II, Src: AAEONTechnol_96:47:f1 (00:07:32:96:47:f1), Dst: BelkinIntern_74:47:63 (d8:ec:5e:74:47:63)
> Internet Protocol Version 4, Src: 192.168.7.175, Dst: 192.168.7.141
> Transmission Control Protocol, Src Port: 3000, Dst Port: 64786, Seq: 351, Ack: 448, Len: 5
> [2 Reassembled TCP Segments (355 bytes): #23709(350), #23731(5)]
v Hypertext Transfer Protocol, has 2 chunks (including last chunk)
  > HTTP/1.1 200 OK\r\n
    X-Powered-By: Express\r\n
    Content-Type: application/json\r\n
    set-cookie: connect.sid=s%3AEdIJusj-vUOPdL5Kj8V04meSEQ3fdiRs.anmhs69qC537kyj5Zama5VWxzfBKDe913nO1DQ%2BXnwU; Path=/; Expires=Tue, 06 Aug 2024 13:32:03 GMT; HttpOnl…
    Date: Tue, 23 Jul 2024 13:32:03 GMT\r\n
    Connection: keep-alive\r\n
    Transfer-Encoding: chunked\r\n
    \r\n
    [Request in frame: 23688]
    [Time since request: 0.072376000 seconds]
```

Hypertext Transfer Protocol: Protocol | Packets: 31072 · Displayed: 111 (0.4%) | Profile: Default

5G Core Node Connectivity – Service-Based Interfaces

- SBI uses a Service-Based Architecture NFs expose services that other NFs can consume.
- HTTP/2 over TCP/TLS is the primary protocol used for communication.
- Enables efficient, multiplexed, and secure signaling between NFs.
- SBI uses point-to-point communication between specific NFs in the 5G Core.
- Ensures backward compatibility with LTE/EPC for seamless interworking.
- Utilizes GTP-U for user plane data transfer between UPFs.

Reference Architecture Based Interfaces: Service Architecture Based is the foundation of 5G Core, certain interfaces still rely on Reference-Based Communication, particularly when interacting with legacy systems or non-SBA functions. These interfaces follow traditional point-to-point communication models with predefined protocols.

INTERFACE NAME	CONNECTING NODES
Uu	UE and RAN
N1	UE and AMF
N2	RAN and AMF
N3	RAN and UPF
N4	SMF and UPF
N5	PCF and AF
N6	UPF and DN
N7	SMF and PCF
N8	AMF and UDM
N9	UPF and UPF
N10	SMF and UDM
N11	AMF and SMF
N12	AMF and AUSF
N13	AUSF and UDM
N14	AMF and AMF
N15	AMF and PCF

Access and Mobility Management Function (AMF)

Objective: Check Service Status of AMF and its functionality

The **Access and Mobility Management Function (AMF)** is a control-plane network function in the 5G Core (5GC). It handles:

- **Access Management**: Manages UE registration and connection setup.
- **Mobility Management**: Tracks UE location and supports handovers.
- **Authentication Coordination**: Interfaces with AUSF and UDM for UE authentication.
- **Signalling**: Manages NAS signalling over N1/N2 interfaces.

Lab Experiment: Checking AMF Service Status

1. Verify AMF Startup Logs

- If using Open5GS or free5GC, inspect amf.log:
- [AMF] Service started on port 7777
- [AMF] Registered with NRF
- [AMF] Received UE registration request

2. Check NRF Registration

- AMF must register with the **Network Repository Function (NRF)**
- Use NRF dashboard or logs to confirm AMF is listed as an active NF

3. Send Test Registration Request

```
curl -k https://<AMF_IP>:7777/namf-comm/v1/ue-contexts
```

- A valid response confirms AMF is active and reachable.

4. Wireshark Trace Analysis

- Capture NAS signalling between gNB and AMF over N2 interface.

- Confirm:
    - Initial UE message
    - Authentication request/response
    - Registration complete

Procedure	**Expected Output:**
Restarting AMF for registration	Active: active (running)
systemctl restart coral5gs-amd.service	Service ID: 1b87483a-305b-41e9-b670-2985996f8551
AMF registration with NRF: **1.1 AMF request to NRF**	
HEADERS[1]: ***PUT /nnrf-nfm/v1/nf-instances/ 2a92ad60-a017-41ef-96bd-15f0b46f6729***	
Header: 3gpp-sbi-sender-timestamp: Mon, 11 Nov 2024 10:24:43.073 GMT	
Header: 3gpp-sbi-max-rsp-time: 10000	
1.2 NRF response to AMF	
HEADERS[19]: 201 Created Header: :status: 201 Created	Value: 201 : status: 201
Json body: **DATA[305], JSON (application/j son-p atch+json)**	
Object Member: nfInstanceId	Member: nfInstanceId

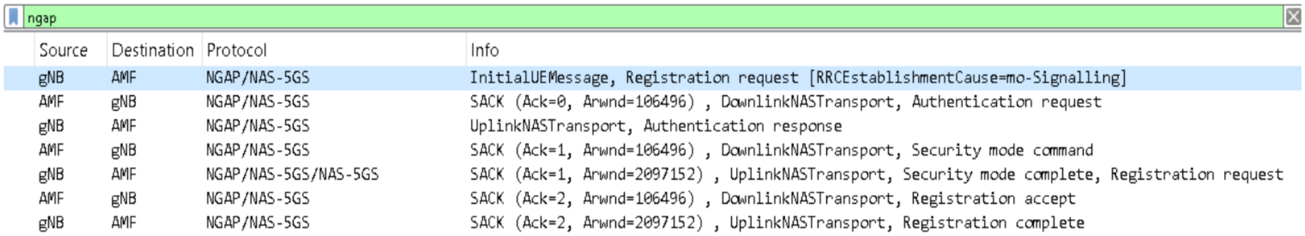
	[Path with value: /nfInstanceId:2a92ad60-a017-41ef-96bd-15f0b46f6729] [Member with value: nfInstanceId:2a92ad60-a017-41ef-96bd-15f0b46f6729] String value: 2a92ad60-a017-41ef-96bd-15f0b46f6729 Key: nfInstanceId [Path: /nfInstanceId]
Member: nfType	Member: nfType [Path with value: /nfType:AMF] [Member with value: nfType:AMF] String value: AMF Key: nfType [Path: /nfType]
Member: nfStatus	Member: nfStatus [Path with value: /nfStatus:REGISTERED] [Member with value: nfStatus:REGISTERED] String value: REGISTERED Key: nfStatus [Path: /nfStatus]
Member: nfProfileChangesInd	
V**erification of Communication service API: -**	
UE Registration **Registration request from UE to AMF on N1 Interface:** **InitialUEMessage, Registration request,**	
[RRCEstablishmentCause=mo-Signalling]	RAN-UE-NGAP-ID 85
protocolIEs: 5 items	
Item 0: id-RAN-UE-NGAP-ID	nR-CGI

	pLMNIdentity: 00f110
Item 1: id-NAS-PDU	
	Mobile Country Code (MCC): Unknown (001)
	Mobile Network Code (MNC): Unknown (01)
Item 2: id-UserLocationInformation	0000 0000 0000 0000 0000 0000 0000 0000 1111.... = nRCellIdentity: 0x00000000ftAI
	pLMNIdentity: 00f110
	Mobile Country Code (MCC): Unknown (001)
	Mobile Network Code (MNC): Unknown (01)
	tAC: 1 (0x000001)
Item 3: id-RRCEstablishmentCause	
Item 4: id-UEContextRequest	
Registration accept from AMF to UE	
initialContextSetupRequest, Registration accept	AMF-UE-NGAP-ID: 1
protocolIEs: 8 items	RAN-UE-NGAP-ID: 41
Item 0: id-AMF-UE-NGAP-ID	
Item 1: id-RAN-UE-NGAP-ID	s-NSSAI
	sST: 01
Item 2: id-GUAMI Item	sD: 030609
3: id-AllowedNSSAI	
	SecurityKey: a81a1340f7ff77 cb880f3872527536c515fe4fc1f91f27
	eea6f45729bb02ccb4 [bit length 256]
Item 4: id-UESecurityCapabilities	
	.... .001 = 5GS registration result: 3GPP access (1)
Item 5: id-SecurityKey	

Item 6: id-MaskedIMEISV Item 7: id-NAS-PDU **Registration complete from UE to AMF** **UplinkNASTransport,Registration complete protocolIEs: 4 items** Item 0: id-AMF-UE-NGAP-ID Item 1: id-RAN-UE-NGAP-ID Item 2: id-NAS-PDU Item 3: id-UserLocationInformation	AMF-UE-NGAP-ID: 1 RAN-UE-NGAP-ID: 41 Message type: Registration complete (0x43) tAC: 1 (0x000001)

Analyzing in Wireshark:

Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on NGAP

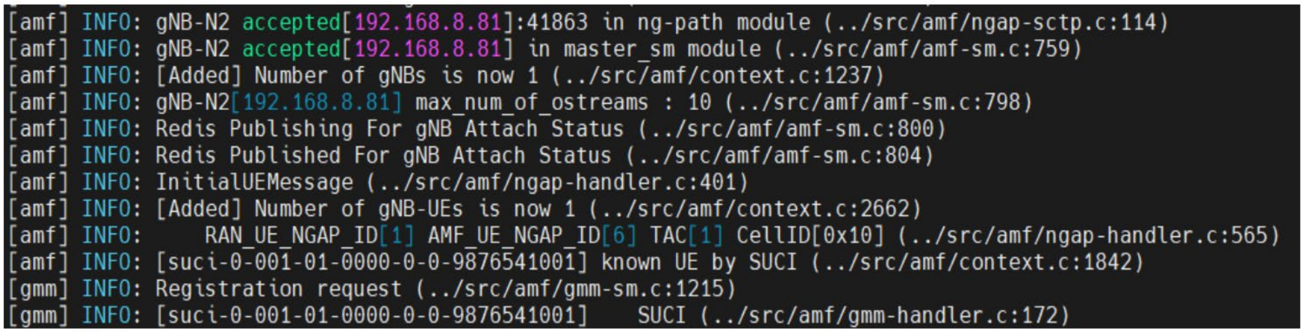


```
ngap
Source  Destination  Protocol               Info
gNB     AMF          NGAP/NAS-5GS           InitialUEMessage, Registration request [RRCEstablishmentCause=mo-Signalling]
AMF     gNB          NGAP/NAS-5GS           SACK (Ack=0, Arwnd=106496) , DownlinkNASTransport, Authentication request
gNB     AMF          NGAP/NAS-5GS           UplinkNASTransport, Authentication response
AMF     gNB          NGAP/NAS-5GS           SACK (Ack=1, Arwnd=106496) , DownlinkNASTransport, Security mode command
gNB     AMF          NGAP/NAS-5GS/NAS-5GS   SACK (Ack=1, Arwnd=2097152) , UplinkNASTransport, Security mode complete, Registration request
AMF     gNB          NGAP/NAS-5GS           SACK (Ack=2, Arwnd=106496) , DownlinkNASTransport, Registration accept
gNB     AMF          NGAP/NAS-5GS           SACK (Ack=2, Arwnd=2097152) , UplinkNASTransport, Registration complete
```

Checking real-time logs for AMF on putty in 5G Core:

Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/amf.log**

AMF log in 5G core on putty terminal



```
[amf] INFO: gNB-N2 accepted[192.168.8.81]:41863 in ng-path module (../src/amf/ngap-sctp.c:114)
[amf] INFO: gNB-N2 accepted[192.168.8.81] in master_sm module (../src/amf/amf-sm.c:759)
[amf] INFO: [Added] Number of gNBs is now 1 (../src/amf/context.c:1237)
[amf] INFO: gNB-N2[192.168.8.81] max_num_of_ostreams : 10 (../src/amf/amf-sm.c:798)
[amf] INFO: Redis Publishing For gNB Attach Status (../src/amf/amf-sm.c:800)
[amf] INFO: Redis Published For gNB Attach Status (../src/amf/amf-sm.c:804)
[amf] INFO: InitialUEMessage (../src/amf/ngap-handler.c:401)
[amf] INFO: [Added] Number of gNB-UEs is now 1 (../src/amf/context.c:2662)
[amf] INFO:     RAN_UE_NGAP_ID[1] AMF_UE_NGAP_ID[6] TAC[1] CellID[0x10] (../src/amf/ngap-handler.c:565)
[amf] INFO: [suci-0-001-01-0000-0-0-9876541001] known UE by SUCI (../src/amf/context.c:1842)
[gmm] INFO: Registration request (../src/amf/gmm-sm.c:1215)
[gmm] INFO: [suci-0-001-01-0000-0-0-9876541001]    SUCI (../src/amf/gmm-handler.c:172)
```

Authentication Server Function (AUSF)

Objective: Check Service Status of AUSF and its functionality

To **check the service status of AUSF** and understand its **functionality in a 5G network**, here's a structured breakdown based on current standards and implementations:

What Is AUSF in 5G?

The **Authentication Server Function (AUSF)** is a key control-plane network function in the 5G Core (5GC). It handles:

- **UE Authentication**: Validates subscriber identity via 5G AKA or EAP-AKA'.
- **Key Derivation**: Generates cryptographic keys for secure signalling and user data.
- **Interface Management**: Communicates with AMF and UDM via the **Service-Based Interface (SBI)**.

More details are available in Telecom Trainer's AUSF overview.

How to Check AUSF Service Status

1. Log Inspection (e.g., Open5GS or free5GC)

- Check ausf.log for entries like:
- [AUSF] Service started on port 7777
- [AUSF] Received authentication request from AMF
- Look for successful authentication vector exchanges and session establishment.

2. Service-Based Interface (SBI) Health

- AUSF listens for requests via HTTPS on a defined port (e.g., 7777).
- Use curl or Postman to send a test request:

  ```
  curl -k https://<AUSF_IP>:7777/nausf-auth/v1/ue-authentications
  ```

- A valid response confirms service availability.

3. NRF Registration

- AUSF must register with the **Network Repository Function (NRF)**.
- Check NRF logs or dashboard to confirm AUSF is listed as an active NF.

4. Authentication Flow Verification

- Use Wireshark or log traces to confirm:
    - AMF → AUSF request
    - AUSF → UDM for AV (Authentication Vector)
    - AUSF → AMF response with challenge

Procedure	Expected Output
Run command systemctl restart coral5gsausfd.service OR by Wireshark	Active: active (running) Service ID: f069e640-304a-41e9-8bcef30e3795f078
AUSF registration with NRF **AUSF request to NRF:** HEADERS [4747]: PUT ***/nnrf-nfm/v1/nfinstances/b9f1a6ac-9ff7-41ef-b9b5-c12304175775***	Value: AUSF user-agent: AUSF Value: NRF
Header: ***user-agent: AUSF*** Header: 3gpp-sbisendertimestamp: Mon, 11 Nov 202410:24:43.073 GMT Header: 3gpp-sbi-maxrsp-time: 10000	
NRF response to AUSF: **HEADERS[4747]: 201 Created** Header: :status: 201 Created	Value: 201 : status: 201
Json Body: **DATA[4747], JSON (application/j son)**	

Object Member: nfInstanceId Member: nfType	Member: nfType [Path with value: /nfType:AUSF] [Member with value: nfType:AUSF] String value: AUSF Key: nfType [Path: /nfType]
Member: nfStatus	Member: nfStatus [Path with value: /nfStatus:REGISTERED] [Member with value: nfStatus:REGISTERED] String value: REGISTERED Key: nfStatus [Path: /nfStatus]
Member: nfProfileChangesSupportInd	Value: nausf-auth
APIs 1. **Service provided by AUSF** **url :*POST /nausf-auth/v1/ueauthentications*** **SBI Interface Reference Point - N12(Between AUSF and AMF)** Header: 3gpp-sbi-sendertimestamp Header: 3gpp-sbi-maxrsp-time: 10000 **Json body:** **DATA[3], JSON (application/json)** **Object** Member: supiOrSuci	Member : supiOrSuci [Path with value: /supiOrSuci:suci-0-001-01-0-

Member: servingNetworkName **2.SBI Interface Reference APIs** **API - *POST /nudmueau/v1/suci-0-001-01-0-0-0-0123456902/securityinformation/generateauth-data* (Between AUSF and UDM)** Method-POST Interface Reference Point - N13 Header: 3gpp-sbi-sendertimestamp: Mon, 11 Nov 2024 10:24:43.073 GMT Header: user-agent: AUSF	0-0-0123456902] [Member with value: supiOrSuci:suci-0-001-01-0-0-0-0123456902] String value: suci-0-001-01-0-0-0-0123456902 Key: supiOrSuci [Path: /supiOrSuci] [Path with value: /servingNetworkName:5G:mnc001.mcc001.3gppnetwork.org] [Member with value: servingNetworkName:5G:mnc001.mcc001.3gppnetwork.org] String value: 5G:mnc001.mcc001.3gppnetwork.org Value: nudm-ueau Value: AUSF user-agent: AUSF Value: UDM

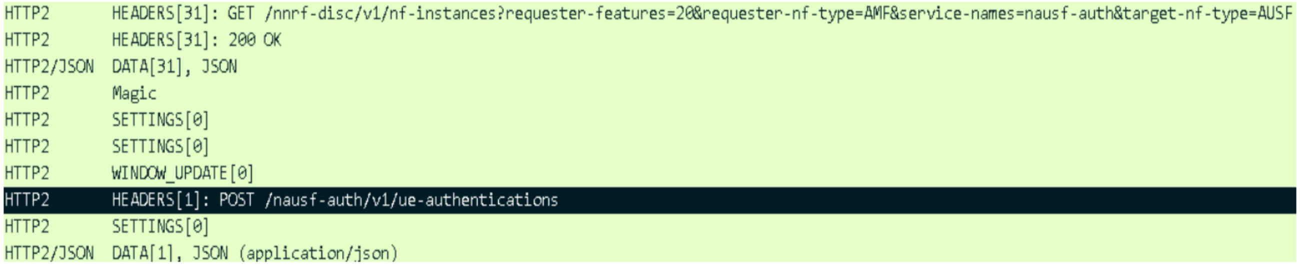
Header: 3gpp-sbi-maxrsp-time: 10000	
Response from UDM to AUSF	
HEADERS[707]: 200 OK Header: :status: 200 OK	Value: 200 :status: 200
DATA[13], JSON DOT (application/j son)	
Object **Member:** authType	String value: 5G_AKA Key: authTyp
Member: authenticationVector	Member: rand [Path with value: /authenticationVector/rand:a141508579f08b5c4d9d76681490cdf7] [Member with value: rand:a141508579f08b5c4d9d76681490cdf7]String value: a141508579f08b5c4d9d76681490cdf7 Key: rand [Path: /authenticationVector/rand] Member: autn [Path with value: /authenticationVector/autn:062409cf9e1e8000c55bff42880e2351][Member with value autn:062409cf9e1e8000c55bff42880e2351]String value: 062409cf9e1e8000c55bff42880e2351 Key:

Member: supi	autn [Path: /authenticationVector/autn] Member: kausf [Path with value: /authenticationVector/kausf:93718cc4d195 c9ee 5c8ad05ee89cfd9af03c35830ef9193008aa1 515 e21482da] [Member with value: kausf:93718cc4d195c9ee5c8ad05ee89cfd9 af03 c35830ef9193008aa1515e21482da] String value: 93718cc4d195c9ee5c8ad05ee89cfd9af03c3 583 0ef9193008aa1515e21482da Key: kausf [Path: /authenticationVector/kausf] Member: supi [Path with value: /supi:imsi- 001010123456902] [Member with value: supi:imsi- 001010123456902] String value: imsi-001010123456902 Key: supi [Path: /supi]
Response from AUSF to AMF **HEADERS[3]: 201 Created Header: :status: 201 Created** **DATA[23], JSON (application/3 gpphal+json)** **Object** Member: authType	Value: 201 : status: 201
Member:5gAuthData	Member: authType [Path with value: /authType:5G_AKA] [Member with value: authType:5G_AKA]String value:

Member: links	5G_AKAKey: authType[Path: /authType] Member: rand [Path with value: /5gAuthData/rand:d7d85f51c96c02075784 1fcf e20d3006] [Member with value: rand:d7d85f51c96c020757841fcfe20d3006]String value: d7d85f51c96c020757841fcfe20d3006Key: rand[Path: /5gAuthData/rand] Member: autn [Path with value: /5gAuthData/autn:c804de5f615680009884 57d 7a9740ff0 [Member with value:autn:c804de5f61568000988457d7a9 740f f0]String value: c804de5f61568000988457d7a9740ff0Key: autn[Path: /5gAuthData/autn]

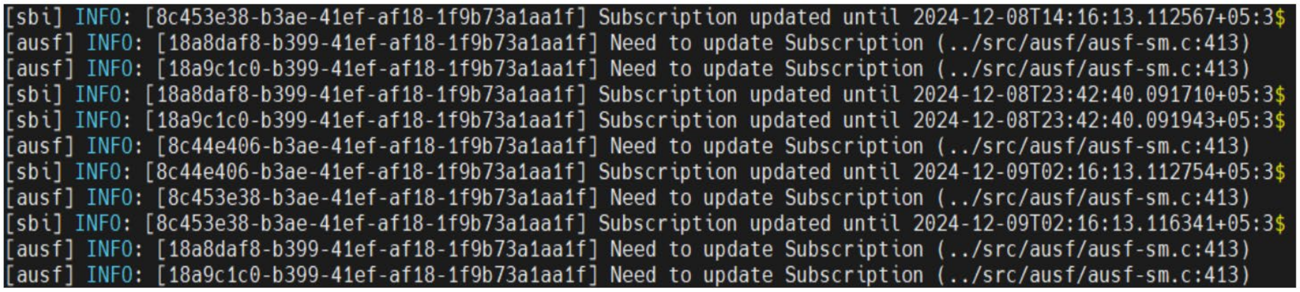
Analyzing in Wireshark:

Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on HTTP2



```
HTTP2       HEADERS[31]: GET /nnrf-disc/v1/nf-instances?requester-features=20&requester-nf-type=AMF&service-names=nausf-auth&target-nf-type=AUSF
HTTP2       HEADERS[31]: 200 OK
HTTP2/JSON  DATA[31], JSON
HTTP2       Magic
HTTP2       SETTINGS[0]
HTTP2       SETTINGS[0]
HTTP2       WINDOW_UPDATE[0]
HTTP2       HEADERS[1]: POST /nausf-auth/v1/ue-authentications
HTTP2       SETTINGS[0]
HTTP2/JSON  DATA[1], JSON (application/json)
```

Checking real-time logs for AUSF on putty in 5G Core:
Run command: $ tail -f /opt/coral5gs/var/log/coral5gs/ausf.log
AUSF log in 5G core on putty terminal



```
[sbi] INFO: [8c453e38-b3ae-41ef-af18-1f9b73a1aa1f] Subscription updated until 2024-12-08T14:16:13.112567+05:3$
[ausf] INFO: [18a8daf8-b399-41ef-af18-1f9b73a1aa1f] Need to update Subscription (../src/ausf/ausf-sm.c:413)
[ausf] INFO: [18a9c1c0-b399-41ef-af18-1f9b73a1aa1f] Need to update Subscription (../src/ausf/ausf-sm.c:413)
[sbi] INFO: [18a8daf8-b399-41ef-af18-1f9b73a1aa1f] Subscription updated until 2024-12-08T23:42:40.091710+05:3$
[sbi] INFO: [18a9c1c0-b399-41ef-af18-1f9b73a1aa1f] Subscription updated until 2024-12-08T23:42:40.091943+05:3$
[ausf] INFO: [8c44e406-b3ae-41ef-af18-1f9b73a1aa1f] Need to update Subscription (../src/ausf/ausf-sm.c:413)
[sbi] INFO: [8c44e406-b3ae-41ef-af18-1f9b73a1aa1f] Subscription updated until 2024-12-09T02:16:13.112754+05:3$
[ausf] INFO: [8c453e38-b3ae-41ef-af18-1f9b73a1aa1f] Need to update Subscription (../src/ausf/ausf-sm.c:413)
[sbi] INFO: [8c453e38-b3ae-41ef-af18-1f9b73a1aa1f] Subscription updated until 2024-12-09T02:16:13.116341+05:3$
[ausf] INFO: [18a8daf8-b399-41ef-af18-1f9b73a1aa1f] Need to update Subscription (../src/ausf/ausf-sm.c:413)
[ausf] INFO: [18a9c1c0-b399-41ef-af18-1f9b73a1aa1f] Need to update Subscription (../src/ausf/ausf-sm.c:413)
```

Data Network (DN)

Objective: Check Service Status of DN-Internet access by 3rd party service and its functionality

Procedure	Expected Output
Ping request from UE to DN: **Echo (ping) request id=0x002a, seq=2/512, ttl=64 (reply in 11)** GPRS Tunneling Protocol TEID: 0x00002f82 (12162) Internet Protocol Version 4, Src: 10.101.0.x (10.101.0.x), Dst: dns.google (8.8.8.8) Source Address: 10.101.0.x (10.101.0.x) Destination Address: dns.google (8.8.8.8) [Stream index: 0] **Ping reply from DN to UE:** **Echo (ping) reply id=0x002a, seq=2/512, ttl=55 (request in 6)** GPRS Tunnelling Protocol Flags: 0x34 Message Type: T-PDU (0xff) Length: 92 TEID: 0x00000001 (1) Source Address: dns.google (8.8.8.8)	TEID: 0x00002f82 (12162) Source Address: 10.101.0.x Destination Address: dns.google (8.8.8.8) TEID: 0x00000001 (1) Source Address: dns.google (8.8.8.8) Destination Address: 10.101.0.x (101.202.0.2)

Destination Address: 10.101.0.x (10.101.0.x) [Stream index: 0]	

Network Exposure Function (NEF)

Objective: Check Service Status of NEF and its functionality

The Network Exposure Function (NEF) is a key component of the 5G Core Network that enables secure and controlled exposure of network services and capabilities to external applications, systems, and third-party developers.

- It acts as a gateway between internal 5G network functions and external Application Functions (AFs).
- NEF is part of the Service-Based Architecture (SBA) defined by 3GPP for 5G.

Procedure	Expected Output
Run command $ core-network status nef	**Expected Output:** Service: coral5gs-nefd.service Active: active (running) Service ID: 1b87483a-305b-41e9-b670-2985996f8551 The service ID in the example provided is `1b87483a-305b-41e9-b670-2985996f8551`. Your specific service ID may vary depending on your system or setup.

Explanation:

- **Service**: Displays the service name of the service for NEF the service name is: coral5gs-nefd.service.
- **Active**: Indicates the status as active (running).
- **Service ID**: Shows the specific Service ID 1b87483a-305b-41e9-b670-2985996f8551.

You can find more details by searching for the Service ID within the captured packets

Network Repository Function (NRF)

The Network Repository Function (NRF) is a central component of the 5G Core Network that maintains a dynamic registry of all available Network Functions (NFs) and their capabilities.

- It enables service discovery, registration, and inter-NF communication within the Service-Based Architecture (SBA) of 5G.
- Think of it as the "directory service" or "yellow pages" of the 5G core.

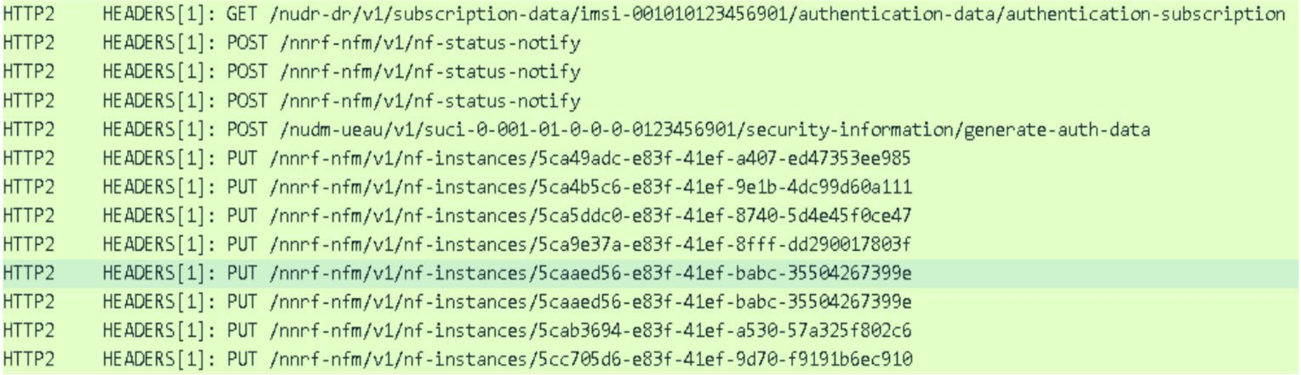
Objective: Check Service Status of NRF and its functionality

Procedure	Expected Output
Run command Systemctl restart coral5gs-nrfd.service NF discovery service triggering by AMF for AUSF to NRF: HEADERS[105]: GET ***/nnrf-disc/v1/nf-instances?requester-features=20&requester-nf-type=AMF&service-names=nausf-auth&target-nf-type=AUSF*** Header: :path: /nnrf-disc/v1/nf-instances?requester-features=20&requester-nf-type=AMF&service-names=nausf-auth&target-**nf-type=AUSF** **Response from NRF** HEADERS[105]: 200 OK **Json body:** **DATA[471], JSON (application/json)** Object Member: validityPeriod Member: nfInstances Array Object Member: nfInstanceId Member: nfType Member: nfStatus	**Expected Output:** Service: coral5gs-nrfd.service Active: active (running) Value: /nnrf-disc/v1/nf-instances?requester-features=20&requester-nf-type=AMF&service-names=nausf-auth&target-nf-type=AUS

Member: heartBeatTimer Member: ipv4Addresses Member: nfServiceList	HEADERS[105]: 200 OK Member: serviceName [Path with value: /nfInstances/[]/nfServiceList/33ff5732-a00c-41ef-b14f-f5dec28a6926/serviceName:nausf-auth] [Member with value: serviceName:nausf-auth] String value: nausf-auth Key: serviceName [Path: /nfInstances/[]/nfServiceList/33ff5732-a00c-41ef-b14f-f5dec28a6926/serviceName]

Analyzing in Wireshark:

Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on HTTP2

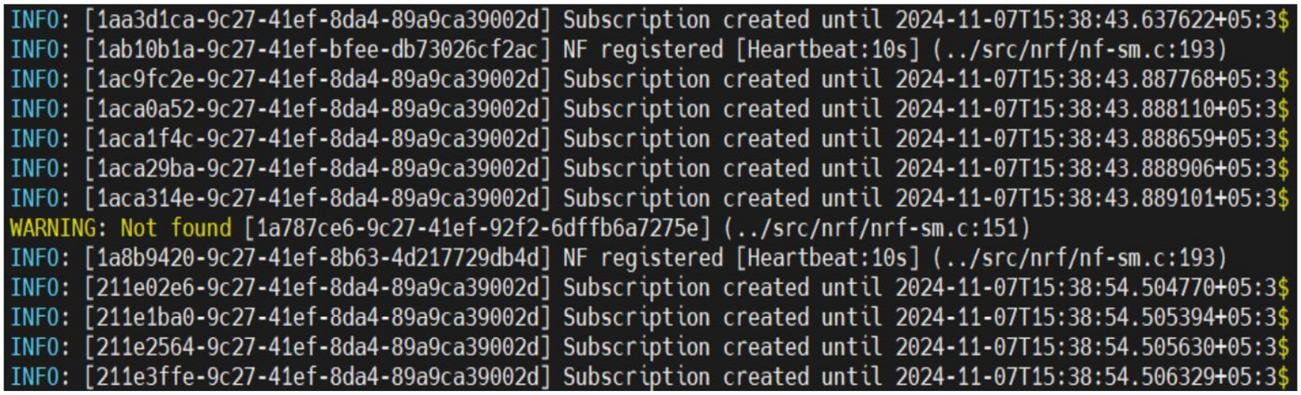


```
HTTP2    HEADERS[1]: GET /nudr-dr/v1/subscription-data/imsi-001010123456901/authentication-data/authentication-subscription
HTTP2    HEADERS[1]: POST /nnrf-nfm/v1/nf-status-notify
HTTP2    HEADERS[1]: POST /nnrf-nfm/v1/nf-status-notify
HTTP2    HEADERS[1]: POST /nnrf-nfm/v1/nf-status-notify
HTTP2    HEADERS[1]: POST /nudm-ueau/v1/suci-0-001-01-0-0-0-0123456901/security-information/generate-auth-data
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5ca49adc-e83f-41ef-a407-ed47353ee985
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5ca4b5c6-e83f-41ef-9e1b-4dc99d60a111
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5ca5ddc0-e83f-41ef-8740-5d4e45f0ce47
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5ca9e37a-e83f-41ef-8fff-dd290017803f
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5caaed56-e83f-41ef-babc-35504267399e
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5caaed56-e83f-41ef-babc-35504267399e
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5cab3694-e83f-41ef-a530-57a325f802c6
HTTP2    HEADERS[1]: PUT /nnrf-nfm/v1/nf-instances/5cc705d6-e83f-41ef-9d70-f9191b6ec910
```

Checking real-time logs for NRF on putty in 5G Core:

Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/nrf.log**

NRF log in 5G core on putty terminal



```
INFO: [1aa3d1ca-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:43.637622+05:3$
INFO: [1ab10b1a-9c27-41ef-bfee-db73026cf2ac] NF registered [Heartbeat:10s] (../src/nrf/nf-sm.c:193)
INFO: [1ac9fc2e-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:43.887768+05:3$
INFO: [1aca0a52-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:43.888110+05:3$
INFO: [1aca1f4c-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:43.888659+05:3$
INFO: [1aca29ba-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:43.888906+05:3$
INFO: [1aca314e-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:43.889101+05:3$
WARNING: Not found [1a787ce6-9c27-41ef-92f2-6dffb6a7275e] (../src/nrf/nrf-sm.c:151)
INFO: [1a8b9420-9c27-41ef-8b63-4d217729db4d] NF registered [Heartbeat:10s] (../src/nrf/nf-sm.c:193)
INFO: [211e02e6-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:54.504770+05:3$
INFO: [211e1ba0-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:54.505394+05:3$
INFO: [211e2564-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:54.505630+05:3$
INFO: [211e3ffe-9c27-41ef-8da4-89a9ca39002d] Subscription created until 2024-11-07T15:38:54.506329+05:3$
```

Network Slice Selection Function (NSSF)

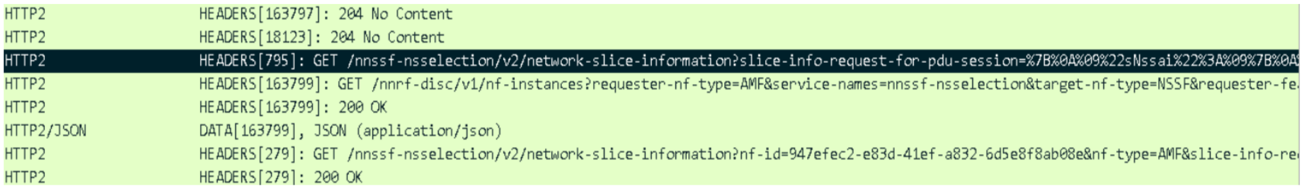
Objective: Check Service Status of NSSF and its functionality

Procedure	Expected Output
Restart NSSF for registration Systemctl restart coral5gs-nssfd.service	**Expected Output:** Service: coral5gs-nssfd.service Active: active (running)
1. NSSF to NRF register: URL: ***/nnrf-nfm/v1/nf-instances/74e5e806-a020-41ef-b2dd-0f1785063434***	Service ID: 1b87483a-305b-41e9-b670-2985996f8551
Header: user-agent: NSSF Header: 3gpp-sbi-sender-timestamp: Mon, 11 Nov 2024 11:31:13.239 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **Response from NRF to NSSF** HEADERS[1]: 201 Created **Json body:** **DATA[1], JSON (application/j son)**	
Object Member: nfInstanceId Member: nfTyp	path: /nnrf-nfm/v1/nf-instances/74e5e806-a020-41ef-b2dd-0f1785063434
	Value: /nnrf-nfm/v1/nf-instances/74e5e806-a020-41ef-b2dd-0f1785063434
Member: nfStatus	
	Value: 201 :status: 201
Member: nfProfileChangesSupportInd	

	Member: nfType [Path with value: /nfType:NSSF] [Member with value: nfType:NSSF] String value: NSSF Key: nfType [Path: /nfType] Member: nfStatus [Path with value: /nfStatus:REGISTERED] [Member with value: nfStatus:REGISTERED] String value: REGISTERED Key: nfStatus [Path: /nfStatus]

Analyzing in Wireshark:

Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on HTTP2



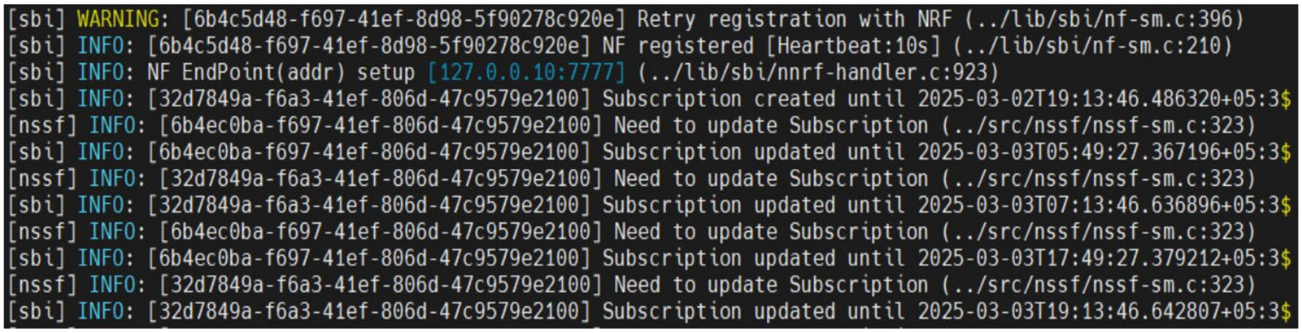
```
HTTP2                HEADERS[163797]: 204 No Content
HTTP2                HEADERS[18123]: 204 No Content
HTTP2                HEADERS[795]: GET /nnssf-nsselection/v2/network-slice-information?slice-info-request-for-pdu-session=%7B%0A%09%22sNssai%22%3A%09%7B%0A%
HTTP2                HEADERS[163799]: GET /nnrf-disc/v1/nf-instances?requester-nf-type=AMF&service-names=nnssf-nsselection&target-nf-type=NSSF&requester-fe
HTTP2                HEADERS[163799]: 200 OK
HTTP2/JSON           DATA[163799], JSON (application/json)
HTTP2                HEADERS[279]: GET /nnssf-nsselection/v2/network-slice-information?nf-id=947efec2-e83d-41ef-a832-6d5e8f8ab08e&nf-type=AMF&slice-info-re
HTTP2                HEADERS[279]: 200 OK
```

HTTP2 messages for NSSF in Wireshark

Checking real-time logs for NSSF on putty in 5G Core:

Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/nssf.log**

NSSF log in 5G core on putty terminal



```
[sbi] WARNING: [6b4c5d48-f697-41ef-8d98-5f90278c920e] Retry registration with NRF (../lib/sbi/nf-sm.c:396)
[sbi] INFO: [6b4c5d48-f697-41ef-8d98-5f90278c920e] NF registered [Heartbeat:10s] (../lib/sbi/nf-sm.c:210)
[sbi] INFO: NF EndPoint(addr) setup [127.0.0.10:7777] (../lib/sbi/nnrf-handler.c:923)
[sbi] INFO: [32d7849a-f6a3-41ef-806d-47c9579e2100] Subscription created until 2025-03-02T19:13:46.486320+05:3$
[nssf] INFO: [6b4ec0ba-f697-41ef-806d-47c9579e2100] Need to update Subscription (../src/nssf/nssf-sm.c:323)
[sbi] INFO: [6b4ec0ba-f697-41ef-806d-47c9579e2100] Subscription updated until 2025-03-03T05:49:27.367196+05:3$
[nssf] INFO: [32d7849a-f6a3-41ef-806d-47c9579e2100] Need to update Subscription (../src/nssf/nssf-sm.c:323)
[sbi] INFO: [32d7849a-f6a3-41ef-806d-47c9579e2100] Subscription updated until 2025-03-03T07:13:46.636896+05:3$
[nssf] INFO: [6b4ec0ba-f697-41ef-806d-47c9579e2100] Need to update Subscription (../src/nssf/nssf-sm.c:323)
[sbi] INFO: [6b4ec0ba-f697-41ef-806d-47c9579e2100] Subscription updated until 2025-03-03T17:49:27.379212+05:3$
[nssf] INFO: [32d7849a-f6a3-41ef-806d-47c9579e2100] Need to update Subscription (../src/nssf/nssf-sm.c:323)
[sbi] INFO: [32d7849a-f6a3-41ef-806d-47c9579e2100] Subscription updated until 2025-03-03T19:13:46.642807+05:3$
```

Policy Control Function (PCF)

The Policy Control Function (PCF) is a core component of the 5G Core (5GC) architecture responsible for policy management and enforcement. It ensures that network behaviour aligns with operator-defined rules for QoS, access control, charging, and user-specific conditions

Objective: Checking Service Status of PCF and its functionality

Procedure	Expected Output
Restart pcf for registration Systemctl restart coral5gs-pcfd.service PCF registration with NRF: PCF To NRF: ***PUT /nnrf-nfm/v1/nf-instances/20e3f556-a0be-41ef-ae1a-114b0ba2858c*** Header: user-agent: PCF Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 06:19:52.783 GMT Header: accept: application/json,application/problem+json **Response: From NRF To PCF** HEADERS[1]: 201 Created **DATA[1], JSON (application/json)** Object Member: nfInstanceId Member: nfType	**Expected Output:** Active: active (running) Service ID: 1b87483a-305b-41e9-b670-2985996f8551
	value: 201 status:201
Member: nfStatus	
Member: nfProfileChangesSupportInd	

Interface: **SBI interface reference point N15 Between PCF and AMF** **Communication service API:** **HEADERS[1]*POST /npcf-am-policy-control/v1/policies*** **Header: user-agent: AMF** Header: accept: application/json,application/problem+json Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 06:49:52.837 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **DATA[1], JSON (application/json)** JavaScript Object Notation: application/json Object Member: request Object Member: notificationUri Member: supi Member: accessType allowedSnssais Member: ueAmbr Object Member: uplink	Member: nfType [Path with value: /nfType:PCF] [Member with value: nfType:PCF] String value: PCF Key: nfType [Path: /nfType] Member: nfStatus [Path with value: /nfStatus:REGISTERED] [Member with value: nfStatus:REGISTERED] String value: REGISTERED Key: nfStatus [Path: /nfStatus]

Member: downlink **Between PCF and SMF on interface N7 to show session related parameter 5QI value:** **HEADERS[1]** ***POST /npcf-smpolicycontrol/v1/sm-policies*** Header: user-agent: SMF Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 06:49:53.362 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **Response:** **Json body:** **DATA[55], JSON (application/json)** Object Member: sessRules Member: 5qi	Member: supi [Path with value: /supi:imsi-001010123456902] [Member with value: supi:imsi-001010123456902] String value: imsi-001010123456902 Key: supi [Path: /supi] Member: accessType [Path with value: /accessType:3GPP_ACCESS] [Member with value: accessType:3GPP_ACCESS] String value: 3GPP_ACCESS Key: accessType [Path: /accessType] Member: uplink [Path with value: /ueAmbr/uplink:1000000 Kbps] [Member with value: uplink:1000000 Kbps] String value: 1000000 Kbps Key: uplink [Path: /ueAmbr/uplink] Member: downlink

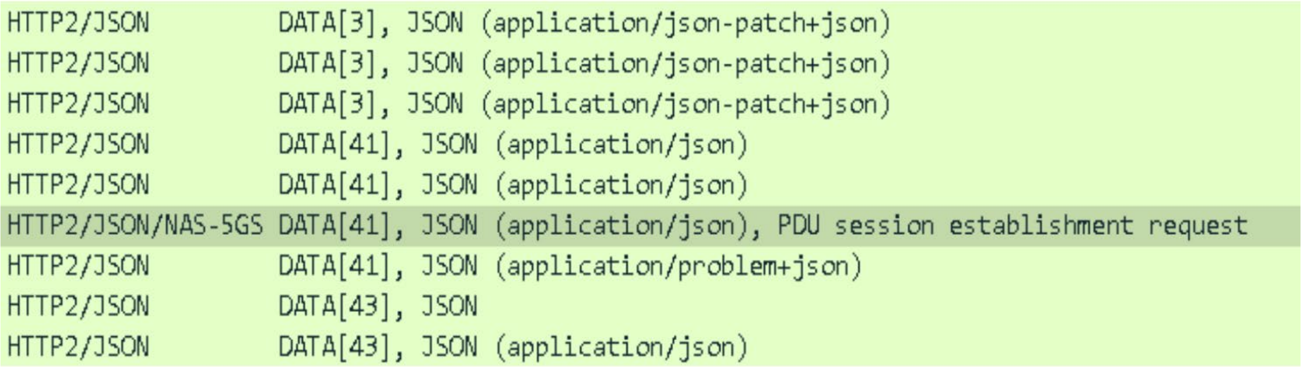
Member: suppFeat	[Path with value: /ueAmbr/downlink:1000000 Kbps] [Member with value: downlink:1000000 Kbps] String value: 1000000 Kbps Key: downlink [Path: /ueAmbr/downlink] **Number value: 9** **Key: 5qi** Member: uplink [Path with value: /subsSessAmbr/uplink:1000000 Kbps]

	[Member with value: uplink:1000000 Kbps] String value: 1000000 Kbps Key: uplink [Path: /subsSessAmbr/uplink] Member: downlink [Path with value: /subsSessAmbr/downlink:1000000 Kbps] [Member with value: downlink:1000000 Kbps] String value: 1000000 Kbps Key: downlink [Path: /subsSessAmbr/downlink] Supported Features: 4000000 ...0 = TSC: False ..0. = ResShare: False .0.. = 3GPP-PS-Data-Off: False 0... = ADC: False ...0 = UMC: False ..0. = NetLoc: False .0.. = RAN-NAS-Cause: False 0... = ProvAFsignalFlow: False ...0 = PCSCF-Restoration-Enhancement: False ..0. = PRA: False .0.. = RuleVersioning: False 0... = SponsoredConnectivity: False ...0 = RAN-Support-Info: False ..0. = PolicyUpdateWhenUESuspends: False .0.. = AccessTypeCondition: False

	0... = MultiIpv6AddrPrefix: False ...0 = SessionRuleErrorHandling: False ..0. = AF_Charging_Identifier: False .0.. = ATSSS: False 0... = PendingTransaction: False ...0 = URLLC: False ..0. = MacAddressRange: False .0.. = WWC: False 0... = QosMonitoring: False ...0 = AuthorizationWithRequiredQoS: False ..0. = EnhancedBackgroundDataTransfer: False .1.. = DN-Authorization: True 0... = PDUSessionRelCause: False

Analyzing in Wireshark:

Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on HTTP2

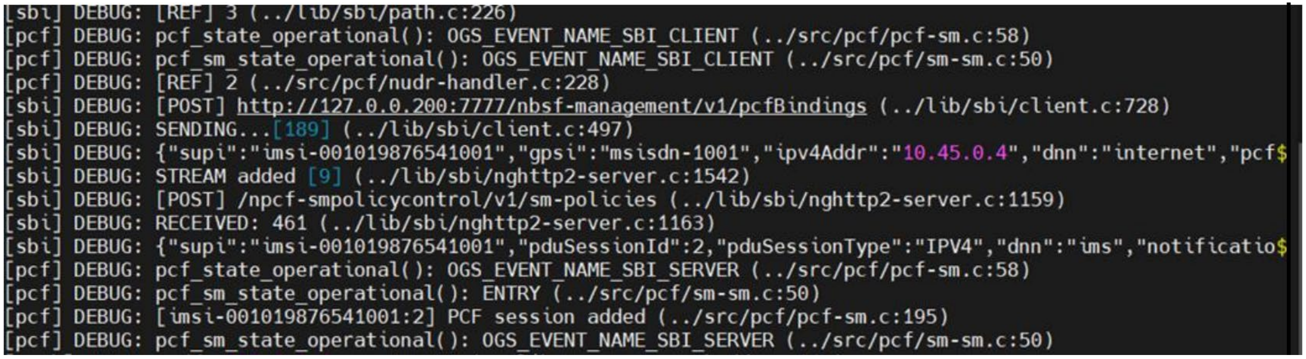


```
HTTP2/JSON         DATA[3], JSON (application/json-patch+json)
HTTP2/JSON         DATA[3], JSON (application/json-patch+json)
HTTP2/JSON         DATA[3], JSON (application/json-patch+json)
HTTP2/JSON         DATA[41], JSON (application/json)
HTTP2/JSON         DATA[41], JSON (application/json)
HTTP2/JSON/NAS-5GS DATA[41], JSON (application/json), PDU session establishment request
HTTP2/JSON         DATA[41], JSON (application/problem+json)
HTTP2/JSON         DATA[43], JSON
HTTP2/JSON         DATA[43], JSON (application/json)
```

Checking real-time logs for PCF on putty in 5G Core:

Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/pcf.log**

PCF log in 5G core on putty terminal



```
[sbi] DEBUG: [REF] 3 (../lib/sbi/path.c:226)
[pcf] DEBUG: pcf_state_operational(): OGS_EVENT_NAME_SBI_CLIENT (../src/pcf/pcf-sm.c:58)
[pcf] DEBUG: pcf_sm_state_operational(): OGS_EVENT_NAME_SBI_CLIENT (../src/pcf/sm-sm.c:50)
[pcf] DEBUG: [REF] 2 (../src/pcf/nudr-handler.c:228)
[sbi] DEBUG: [POST] http://127.0.0.200:7777/nbsf-management/v1/pcfBindings (../lib/sbi/client.c:728)
[sbi] DEBUG: SENDING...[189] (../lib/sbi/client.c:497)
[sbi] DEBUG: {"supi":"imsi-001019876541001","gpsi":"msisdn-1001","ipv4Addr":"10.45.0.4","dnn":"internet","pcf$
[sbi] DEBUG: STREAM added [9] (../lib/sbi/nghttp2-server.c:1542)
[sbi] DEBUG: [POST] /npcf-smpolicycontrol/v1/sm-policies (../lib/sbi/nghttp2-server.c:1159)
[sbi] DEBUG: RECEIVED: 461 (../lib/sbi/nghttp2-server.c:1163)
[sbi] DEBUG: {"supi":"imsi-001019876541001","pduSessionId":2,"pduSessionType":"IPV4","dnn":"ims","notificatio$
[pcf] DEBUG: pcf_state_operational(): OGS_EVENT_NAME_SBI_SERVER (../src/pcf/pcf-sm.c:58)
[pcf] DEBUG: pcf_sm_state_operational(): ENTRY (../src/pcf/sm-sm.c:50)
[pcf] DEBUG: [imsi-001019876541001:2] PCF session added (../src/pcf/pcf-sm.c:195)
[pcf] DEBUG: pcf_sm_state_operational(): OGS_EVENT_NAME_SBI_SERVER (../src/pcf/sm-sm.c:50)
```

Session Management Function (SMF)

The Session Management Function (SMF) is a critical control-plane component in the 5G Core (5GC) architecture. It manages the lifecycle of PDU (Protocol Data Unit) sessions, ensuring efficient data delivery between the User Equipment (UE) and external Data Networks (DN).

Objective: Check Service Status of SMF and its functionality

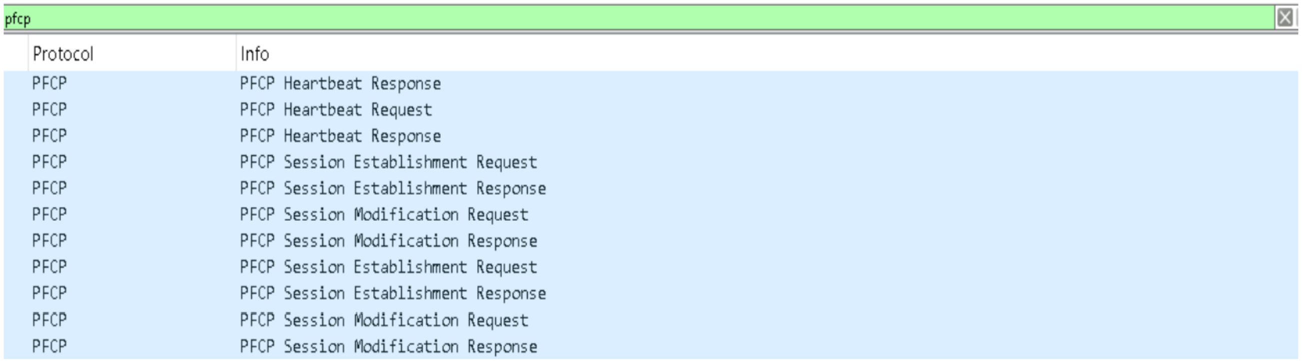
Procedure	Expected Output
Restart SMF for registration Systemctl restart coral5gs-smfd.service **SMF registration with NRF:** **SMF To NRF:** **HEADERS[1]:PUT *****/nnrf-nfm/v1/nf-instances/25565e0a-a0d5-41ef-911b-095b3d943b36*** Header: user-agent: SMF Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 09:04:38.727 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **Response from NRF to SMF** **Json body:** **HEADERS[1]: 201 Created** Object Member: nfInstanceId Member: nfType	**Expected Output:** Service: coral5gs-smfd.service Active: active (running) value: 201
Member: nfStatus	:status:201

Member: nfProfileChangesSupportInd **Service provide by SMF to PCF over N7 interface** **HEADERS[1] POST */npcf-smpolicycontrol/v1/sm-policies*** Header: user-agent: SMF Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 09:47:00.968 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **Response:** **Json body:** **DATA[55], JSON** **(application/json)** Object Member: sessRules Member: 5qi Member: suppFeat **Interface between AMF to SMF on N11** **Request from AMF to SMF For Session context after PDU Session Request Message** **POST** **/nsmf-pdusession/v1/sm-contexts** **Header: user-agent: AMF** Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 09:47:01.073 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **Response from SMF to AMF** **/**namf-comm/v1/ue-contexts/imsi-001010123456902/n1-n2-messages **DATA[1], JSON** **(application/json),**PDU session establishment accept (PDU session type IPv4 only allowed), PDUSessionResourceSetupRequestTransfer Non-Access-Stratum 5GS (NAS)PDU	Member: nfType [Path with value: /nfType:SMF] [Member with value: nfType:SMF] String value: SMF Key: nfType [Path: /nfType] Member: nfStatus [Path with value: /nfStatus:REGISTERED] [Member with value: nfStatus:REGISTERED] String value: REGISTERED Key: nfStatus [Path: /nfStatus]

	Number value: 9 **Key: 5qi** ..00 0001 = Qos flow identifier: 1 Session-AMBR for downlink: 3850 Kbps (3850) Session-AMBR for uplink: 1530 Kbps (1530) DNN: internet.

Analyzing in Wireshark:

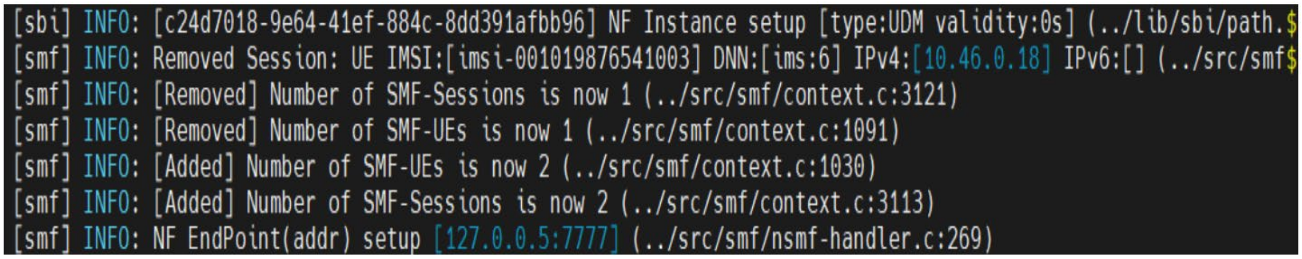
Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on PFCP



pfcp

Protocol	Info
PFCP	PFCP Heartbeat Response
PFCP	PFCP Heartbeat Request
PFCP	PFCP Heartbeat Response
PFCP	PFCP Session Establishment Request
PFCP	PFCP Session Establishment Response
PFCP	PFCP Session Modification Request
PFCP	PFCP Session Modification Response
PFCP	PFCP Session Establishment Request
PFCP	PFCP Session Establishment Response
PFCP	PFCP Session Modification Request
PFCP	PFCP Session Modification Response

Checking real-time logs for SMF on putty in 5G Core:
Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/smf.log**
SMF log in 5G core on putty terminal



```
[sbi] INFO: [c24d7018-9e64-41ef-884c-8dd391afbb96] NF Instance setup [type:UDM validity:0s] (../lib/sbi/path.$
[smf] INFO: Removed Session: UE IMSI:[imsi-001019876541003] DNN:[ims:6] IPv4:[10.46.0.18] IPv6:[] (../src/smf$
[smf] INFO: [Removed] Number of SMF-Sessions is now 1 (../src/smf/context.c:3121)
[smf] INFO: [Removed] Number of SMF-UEs is now 1 (../src/smf/context.c:1091)
[smf] INFO: [Added] Number of SMF-UEs is now 2 (../src/smf/context.c:1030)
[smf] INFO: [Added] Number of SMF-Sessions is now 2 (../src/smf/context.c:3113)
[smf] INFO: NF EndPoint(addr) setup [127.0.0.5:7777] (../src/smf/nsmf-handler.c:269)
```

Unified Data Management (UDM)

The Unified Data Management (UDM) is a core control-plane function in the 5G Core (5GC) architecture responsible for managing subscriber identity, authentication, and access authorization.

- It replaces the Home Subscriber Server (HSS) from 4G and operates within the Service-Based Architecture (SBA).
- UDM works closely with AUSF, PCF, and UDR to ensure secure and policy-compliant access to 5G services.

Objective: Check Service Status of UDM and its functionality

Procedure	Expected Output
Restart UDM for registration Systemctl restart coral5gs-udmd.service **UDM** **1.UDM registration with NRF:** **1.1UDM To NRF:** **URL:** ***/nnrf-nfm/v1/nf-instances/b60e4de0-a0e4-41ef-8d5f-6f64befd3fc7*** Header: user-agent: UDM Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 10:56:03.915 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **1.2 Response from NRF to UDM Stream: HEADERS[1], Stream ID: 1, Length 108, 201 Created** **DATA[1], JSON (application/json)** Object	**Expected Output:** Active: active (running) Service ID: 1b87483a-305b-41e9-b670-2985996f8551

Member: nfInstanceId Member: nfType Member: nfStatus Member: nfProfileChangesSupportInd **2.Service provided by UDM:** **2.1.SBI interface reference point N13: Between AUSF and UDM for authentication related parameters before Authentication request message** **Service : nudm-ueau AUSF TO UDM:** **Stream: HEADERS, Stream ID: 3, Length 75, POST *****/nudmueau/v1/imsi-001010123456901/auth-events*** Header: user-agent: AUSF Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 11:02:42.393 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **UDM RESPONSES TO AUSF:** **Stream: HEADERS, Stream ID: 3, Length 66, 201 Created** **JSON body:** Object Member: nfInstanceId Member: success Member: timeStamp Member: authType	**value: 201:** **status: 201** Member: nfType [Path with value: /nfType:UDM] [Member with value: nfType:UDM] String value: UDM Key: nfType [Path: /nfType] Member: nfStatus [Path with value: /nfStatus:REGISTERED] [Member with value: nfStatus:REGISTERED] String value: REGISTERED Key: nfStatus [Path: /nfStatus]

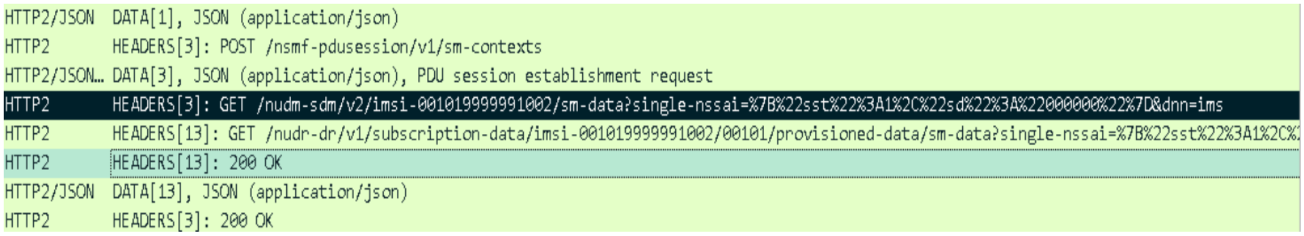
Member: servingNetworkName	
2.2.SBI interface reference point N8: Between AMF and UDM for access related parameters after security mode complete Service:nudm-uecm **AMF to UDM:** **Stream: HEADERS, Stream ID: 5, Length 134, PUT** ***/nudm-uecm/v1/imsi-001010123456902/registrations/amf-3gpp-access*** Header: user-agent: AMF Header: 3gpp-sbi-sender-timestamp: Mon, 11 Nov 2024 10:32:09.013 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **UDM responses to AMF:** **Stream: HEADERS, Stream ID: 1, Length 123, 201 Created**	**Value: 201**
JSON body: Member: mcc	**:status:201**
Member: mnc	Member: success [Path with value: /success:true] [Member with value: success:true]
Member: amfId	True value Key: success [Path: /success] Member: authType [Path with value: /authType:5G_AKA] [Member with value: authType:5G_AKA] String value: 5G_AKA

	Key: authType [Path: /authType] **Value: 201** **:status:201** Member: mcc [Path with value: /guami/plmnId/mcc:001] [Member with value: mcc:001] String value: 001 Key: mcc [Path: /guami/plmnId/mcc] Member: mnc [Path with value: /guami/plmnId/mnc:01] [Member with value: mnc:01] String value: 01 Key: mnc [Path: /guami/plmnId/mnc] Member: amfId

	[Path with value: /guami/amfId:020040] [Member with value: amfId:020040] String value: 020040 Key: amfId [Path: /guami/amfId]

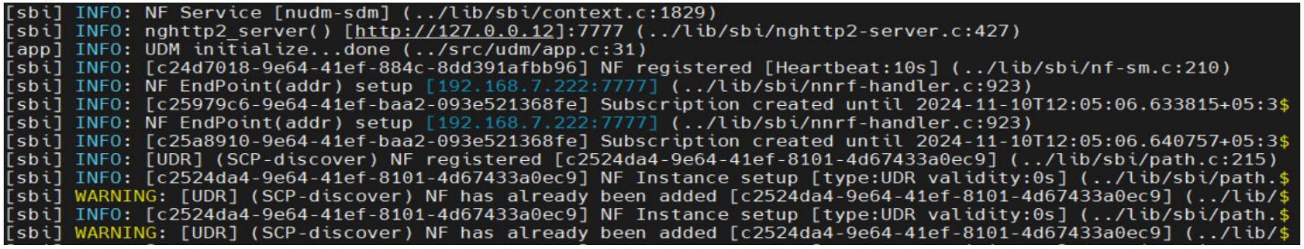
Analyzing in Wireshark:

Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on HTTP@



```
HTTP2/JSON  DATA[1], JSON (application/json)
HTTP2       HEADERS[3]: POST /nsmf-pdusession/v1/sm-contexts
HTTP2/JSON… DATA[3], JSON (application/json), PDU session establishment request
HTTP2       HEADERS[3]: GET /nudm-sdm/v2/imsi-001019999991002/sm-data?single-nssai=%7B%22sst%22%3A1%2C%22sd%22%3A%22000000%22%7D&dnn=ims
HTTP2       HEADERS[13]: GET /nudr-dr/v1/subscription-data/imsi-001019999991002/00101/provisioned-data/sm-data?single-nssai=%7B%22sst%22%3A1%2C%2
HTTP2       HEADERS[13]: 200 OK
HTTP2/JSON  DATA[13], JSON (application/json)
HTTP2       HEADERS[3]: 200 OK
```

Checking real-time logs for UDM on putty in 5G Core:
Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/udm.log**
UDM log in 5G core on putty terminal



```
[sbi] INFO: NF Service [nudm-sdm] (../lib/sbi/context.c:1829)
[sbi] INFO: nghttp2_server() [http://127.0.0.12]:7777 (../lib/sbi/nghttp2-server.c:427)
[app] INFO: UDM initialize...done (../src/udm/app.c:31)
[sbi] INFO: [c24d7018-9e64-41ef-884c-8dd391afbb96] NF registered [Heartbeat:10s] (../lib/sbi/nf-sm.c:210)
[sbi] INFO: NF EndPoint(addr) setup [192.168.7.222:7777] (../lib/sbi/nnrf-handler.c:923)
[sbi] INFO: [c25979c6-9e64-41ef-baa2-093e521368fe] Subscription created until 2024-11-10T12:05:06.633815+05:3$
[sbi] INFO: NF EndPoint(addr) setup [192.168.7.222:7777] (../lib/sbi/nnrf-handler.c:923)
[sbi] INFO: [c25a8910-9e64-41ef-baa2-093e521368fe] Subscription created until 2024-11-10T12:05:06.640757+05:3$
[sbi] INFO: [UDR] (SCP-discover) NF registered [c2524da4-9e64-41ef-8101-4d67433a0ec9] (../lib/sbi/path.c:215)
[sbi] INFO: [c2524da4-9e64-41ef-8101-4d67433a0ec9] NF Instance setup [type:UDR validity:0s] (../lib/sbi/path.$
[sbi] WARNING: [UDR] (SCP-discover) NF has already been added [c2524da4-9e64-41ef-8101-4d67433a0ec9] (../lib/$
[sbi] INFO: [c2524da4-9e64-41ef-8101-4d67433a0ec9] NF Instance setup [type:UDR validity:0s] (../lib/sbi/path.$
[sbi] WARNING: [UDR] (SCP-discover) NF has already been added [c2524da4-9e64-41ef-8101-4d67433a0ec9] (../lib/$
```

Unified Data Repository (UDR)

Unified Data Repository (UDR) is a centralized database that stores and manages various types of user-related and subscription data. It replaces the older concept of "User Data Repository" from previous generations and expands its scope significantly.

Key Functions of UDR

- Centralized Data Storage:
  - Stores subscription data, policy data, structured data for exposure, and application data.
- Supports Multiple Network Functions (NFs):
  - Provides data to NFs like AMF, SMF, AUSF, PCF, and NEF.
  - Acts as a backend for UDM (Unified Data Management), which serves as the front-end interface for accessing subscription data.
- Policy and Application Data Handling:
  - PCF accesses policy data directly from UDR via the N36 interface.
  - NEF enables external Application Functions (AFs) to store and retrieve application-specific data.
- Roaming Support:
  - In roaming scenarios, visited networks can store and manage roamer-specific data locally in their UDRs.
- Security and Isolation:
  - Supports deployment of multiple UDRs for different sets of data or network functions.
  - Enables isolation between environments (e.g., different enterprises or PLMNs).

Objective: Check Service Status of UDR and its functionality

Procedure	Expected Output
Restart UDR for registration Systemctl restart coral5gs-udrd.service **1.UDR registration with NRF** **1.1 UDR To NRF:** **Stream: HEADERS, Stream ID: 1, Length 188, PUT *****/nnrf-nfm/v1/nf-instances/5efce888-a1af-41ef-bff2-91ecfa52ba77*** Header: user-agent: UDR Header: 3gpp-sbi-sender-timestamp: Wed, 13 Nov 2024 11:06:45.677 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **1.2 Response from NRF to UDR** **Stream: HEADERS, Stream ID: 1, Length 107, 201 Created** **JSON body: JavaScript Object Notation: application/jso** Object Member:nfInstanceId Member: nfTyp Member: nfStatus Member: nfProfileChangesSupportInd **2.Service provided by UDR:** **2.1.SBI interface reference point N35:** **Between UDM and UDR for subscription related parameters** **Stream: HEADERS, Stream ID: 45, Length 102, GET *****/nudr-***	**Expected Output:** Service: coral5gs-udrd.service Active: active (running) Service ID: 1b87483a-305b-41e9-b670-2985996f8551 **value:201** **:status:201** Member: nfType [Path with value: /nfType:UDR]

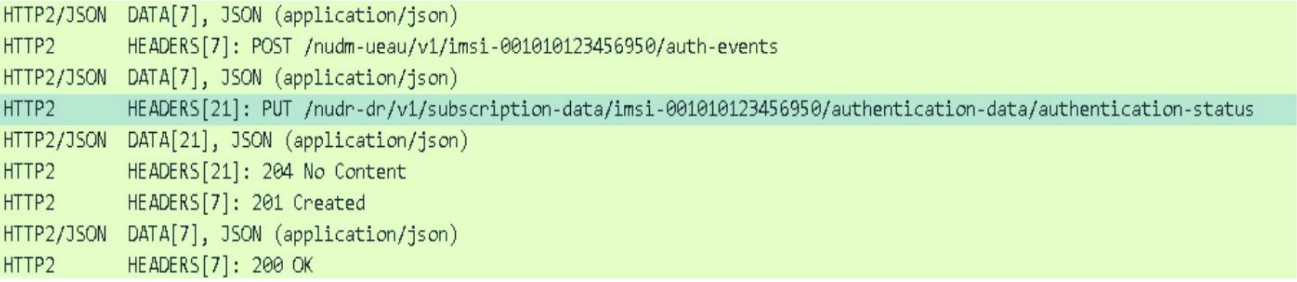
dr/v1/subscription-data/imsi-001010123456902/authentication-data/authentication-subscription Header: user-agent: UDM Header: 3gpp-sbi-sender-timestamp: Tue, 12 Nov 2024 11:02:42.087 GMT Header: 3gpp-sbi-max-rsp-time: 10000 **UDR responses to UDM: HEADERS[1]: 200 OK** **JSON BODY:** **JavaScript Object Notation: application/json** Object Member:authenticationMethod Member: encPermanentKey Member: sequenceNumber Member: encOpcKey	[Member with value: nfType:UDR] String value: UDR Key: nfType [Path: /nfType] Member: nfStatus [Path with value: /nfStatus:REGISTERED] [Member with value: nfStatus:REGISTERED] String value: REGISTERED Key: nfStatus [Path: /nfStatus]

	Value: 200 **:status: 200** Member: authenticationMethod [Path with value: /authenticationMethod:5G_AKA] [Member with value: authenticationMethod:5G_AKA] String value: 5G_AKA Key: authenticationMethod [Path: /authenticationMethod] Member: encPermanentKey [Path with value: /encPermanentKey:11111111111111111111111111111111] [Member with value: encPermanentKey:11111111111111111111111111111111] String value: 11111111111111111111111111111111 Key: encPermanentKey [Path: /encPermanentKey] Member: sequenceNumber Object Member: sqn [Path with value: /sequenceNumber/sqn:000000000841] [Member with value: sqn:000000000841] String value: 000000000841

	Key: sqn [Path: /sequenceNumber/sqn] Key: sequenceNumber [Path: /sequenceNumber] Member: encOpcKey [Path with value: /encOpcKey:11111111111111111111111111111111] [Member with value: encOpcKey:11111111111111111111111111111111] String value: 11111111111111111111111111111111 Key: encOpcKey [Path: /encOpcKey]

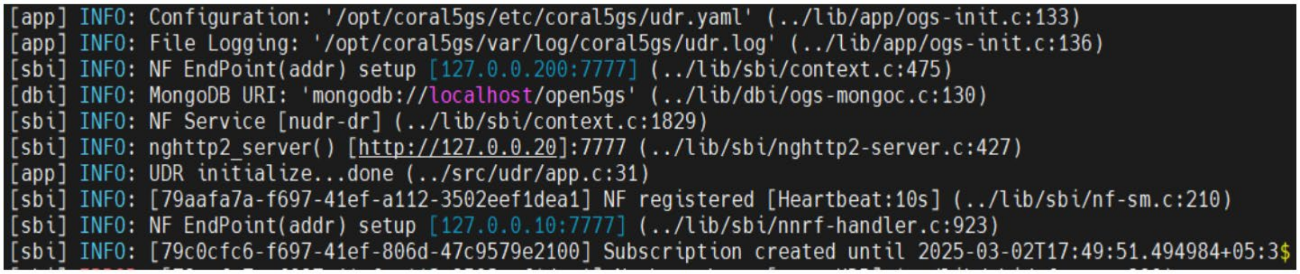
Analyzing in Wireshark:

Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on HTTP2



```
HTTP2/JSON  DATA[7], JSON (application/json)
HTTP2       HEADERS[7]: POST /nudm-ueau/v1/imsi-001010123456950/auth-events
HTTP2/JSON  DATA[7], JSON (application/json)
HTTP2       HEADERS[21]: PUT /nudr-dr/v1/subscription-data/imsi-001010123456950/authentication-data/authentication-status
HTTP2/JSON  DATA[21], JSON (application/json)
HTTP2       HEADERS[21]: 204 No Content
HTTP2       HEADERS[7]: 201 Created
HTTP2/JSON  DATA[7], JSON (application/json)
HTTP2       HEADERS[7]: 200 OK
```

Checking real-time logs for UDR on putty in 5G Core:
Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/udr.log**
UDR log in 5G core on putty terminal



```
[app] INFO: Configuration: '/opt/coral5gs/etc/coral5gs/udr.yaml' (../lib/app/ogs-init.c:133)
[app] INFO: File Logging: '/opt/coral5gs/var/log/coral5gs/udr.log' (../lib/app/ogs-init.c:136)
[sbi] INFO: NF EndPoint(addr) setup [127.0.0.200:7777] (../lib/sbi/context.c:475)
[dbi] INFO: MongoDB URI: 'mongodb://localhost/open5gs' (../lib/dbi/ogs-mongoc.c:130)
[sbi] INFO: NF Service [nudr-dr] (../lib/sbi/context.c:1829)
[sbi] INFO: nghttp2_server() [http://127.0.0.20]:7777 (../lib/sbi/nghttp2-server.c:427)
[app] INFO: UDR initialize...done (../src/udr/app.c:31)
[sbi] INFO: [79aafa7a-f697-41ef-a112-3502eef1dea1] NF registered [Heartbeat:10s] (../lib/sbi/nf-sm.c:210)
[sbi] INFO: NF EndPoint(addr) setup [127.0.0.10:7777] (../lib/sbi/nnrf-handler.c:923)
[sbi] INFO: [79c0cfc6-f697-41ef-806d-47c9579e2100] Subscription created until 2025-03-02T17:49:51.494984+05:3$
```

UDR Logs in 5G core via putty terminal

Unified Data Function (UDF)

In 5G architecture, the User Plane Function (UPF) is a cornerstone of the data plane, responsible for handling user traffic. It plays a vital role in enabling high-speed, low-latency communication and supports advanced features like network slicing and edge computing.

UPF is a network function defined in the 3GPP 5G Core (5GC) architecture. It manages the user plane, which means it deals with actual data packets sent and received by users, unlike the control plane that handles signalling and session management.

Key Responsibilities of UPF

- Packet Routing and Forwarding Routes user data between the Radio Access Network (RAN) and external Data Networks (DN).
- QoS Enforcement Applies Quality of Service rules for uplink and downlink traffic, including rate limiting and packet marking.
- Traffic Inspection and Filtering Performs deep packet inspection and applies traffic steering policies.
- Mobility Anchor Point Serves as an anchor for mobility across different Radio Access Technologies (RATs), such as intra- and inter-RAT handovers.
- Lawful Interception and Usage Reporting Supports lawful intercept and generates usage reports for billing and analytics.
- Edge Deployment Can be deployed close to users (at the edge) to reduce latency and improve performance for applications like AR/VR, autonomous vehicles, and industrial IoT.

Objective: Check Service Status of UPF and its functionality

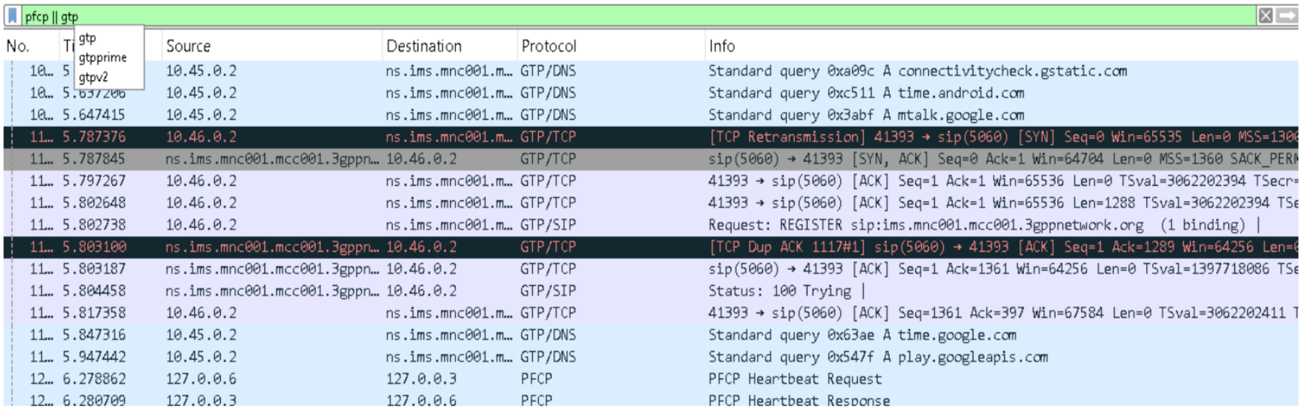
Procedure	Expected Output
Restart UPF for registration Systctemctl restart coral5gs-upfd.service **Registration with SMF:** **Request from SMF side** **PFCP Association Setup Request over N4 interface** Packet Forwarding Control Protocol Message Type: PFCP Association Setup Request (5) Node ID : IPv4 address: 127.0.0.4 **Response from UPF side:** **PFCP Association Setup Response over N4** Packet Forwarding Control Protocol Message Type: PFCP Association Setup Response (6) Node ID : IPv4 address: 127.0.0.7 Cause : Request accepted(success) **Service API:** **Session Connection From SMF side to UPF** **PFCP Session Establishment Request on N4** Packet Forwarding Control Protocol Message Type: PFCP Session Establishment Request (50) PDN Type : IPv4 User ID : APN/DNN : internet **Response from UPF** **PFCP Session Establishment Response on N4** Packet Forwarding Control Protocol Message Type: PFCP Session Establishment Response (51) Node ID : IPv4 address: 127.0.0.7 Cause : Request accepted(success) F-SEID : SEID: 0x0000000000000fc5, IPv4 127.0.0.7 **From SMF to UPF over N4 interface** **PFCP Session Modification Request**	**Expected Output:** Active: active (running) Service ID: 1b87483a-305b-41e9-b670-2985996f8551 IPv4: smf(127.0.0.4) IPv4: upf(127.0.0.7)

Message Type: PFCP Session Modification Request (52) SEID: 0x0000000000000fc5 **PFCP Session Modification Response from UPF to SMF over N4 interface** Message Type: PFCP Session Modification Response (53) Cause : Request accepted(success) **PING service:** **UL packet from UE to DN**	accepted(success) (1) .... .001 = PDN Type: IPv4 (1) IMSI: 001010123456902 APN/DNN: internet

	Request accepted(success) (1) SEID: 0x0000000000000fc5 Cause: Request accepted(success) (1)

Analyzing in Wireshark:

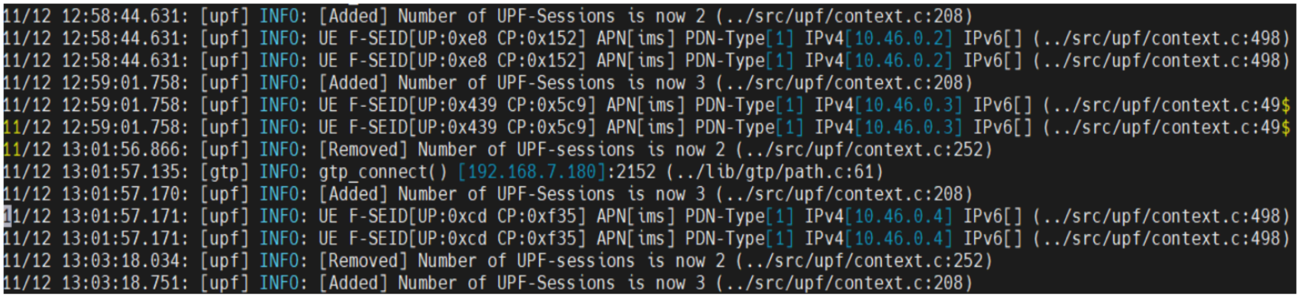
Capture Traffic: Open Wireshark, start capturing packets on the relevant network interface and filter on pfcp and gtp



pfcp || gtp

No.	Ti	Source	Destination	Protocol	Info
10…	5	10.45.0.2	ns.ims.mnc001.m…	GTP/DNS	Standard query 0xa09c A connectivitycheck.gstatic.com
10…	5.637206	10.45.0.2	ns.ims.mnc001.m…	GTP/DNS	Standard query 0xc511 A time.android.com
10…	5.647415	10.45.0.2	ns.ims.mnc001.m…	GTP/DNS	Standard query 0x3abf A mtalk.google.com
11…	5.787376	10.46.0.2	ns.ims.mnc001.m…	GTP/TCP	[TCP Retransmission] 41393 → sip(5060) [SYN] Seq=0 Win=65535 Len=0 MSS=130
11…	5.787845	ns.ims.mnc001.mcc001.3gppn…	10.46.0.2	GTP/TCP	sip(5060) → 41393 [SYN, ACK] Seq=0 Ack=1 Win=64704 Len=0 MSS=1360 SACK_PER
11…	5.797267	10.46.0.2	ns.ims.mnc001.m…	GTP/TCP	41393 → sip(5060) [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=3062202394 TSecr=
11…	5.802648	10.46.0.2	ns.ims.mnc001.m…	GTP/TCP	41393 → sip(5060) [ACK] Seq=1 Ack=1 Win=65536 Len=1288 TSval=3062202394 TS
11…	5.802738	10.46.0.2	ns.ims.mnc001.m…	GTP/SIP	Request: REGISTER sip:ims.mnc001.mcc001.3gppnetwork.org (1 binding) |
11…	5.803100	ns.ims.mnc001.mcc001.3gppn…	10.46.0.2	GTP/TCP	[TCP Dup ACK 1117#1] sip(5060) → 41393 [ACK] Seq=1 Ack=1289 Win=64256 Len=
11…	5.803187	ns.ims.mnc001.mcc001.3gppn…	10.46.0.2	GTP/TCP	sip(5060) → 41393 [ACK] Seq=1 Ack=1361 Win=64256 Len=0 TSval=1397718086 TS
11…	5.804458	ns.ims.mnc001.mcc001.3gppn…	10.46.0.2	GTP/SIP	Status: 100 Trying |
11…	5.817358	10.46.0.2	ns.ims.mnc001.m…	GTP/TCP	41393 → sip(5060) [ACK] Seq=1361 Ack=397 Win=67584 Len=0 TSval=3062202411 T
11…	5.847316	10.45.0.2	ns.ims.mnc001.m…	GTP/DNS	Standard query 0x63ae A time.google.com
11…	5.947442	10.45.0.2	ns.ims.mnc001.m…	GTP/DNS	Standard query 0x547f A play.googleapis.com
12…	6.278862	127.0.0.6	127.0.0.3	PFCP	PFCP Heartbeat Request
12…	6.280709	127.0.0.3	127.0.0.6	PFCP	PFCP Heartbeat Response

Checking real-time logs for UPF on putty in 5G Core:
Run command: **$ tail -f /opt/coral5gs/var/log/coral5gs/upf.log**
UPF log in 5G core on putty terminal



```
11/12 12:58:44.631: [upf] INFO: [Added] Number of UPF-Sessions is now 2 (../src/upf/context.c:208)
11/12 12:58:44.631: [upf] INFO: UE F-SEID[UP:0xe8 CP:0x152] APN[ims] PDN-Type[1] IPv4[10.46.0.2] IPv6[] (../src/upf/context.c:498)
11/12 12:58:44.631: [upf] INFO: UE F-SEID[UP:0xe8 CP:0x152] APN[ims] PDN-Type[1] IPv4[10.46.0.2] IPv6[] (../src/upf/context.c:498)
11/12 12:59:01.758: [upf] INFO: [Added] Number of UPF-Sessions is now 3 (../src/upf/context.c:208)
11/12 12:59:01.758: [upf] INFO: UE F-SEID[UP:0x439 CP:0x5c9] APN[ims] PDN-Type[1] IPv4[10.46.0.3] IPv6[] (../src/upf/context.c:49$
11/12 12:59:01.758: [upf] INFO: UE F-SEID[UP:0x439 CP:0x5c9] APN[ims] PDN-Type[1] IPv4[10.46.0.3] IPv6[] (../src/upf/context.c:49$
11/12 13:01:56.866: [upf] INFO: [Removed] Number of UPF-sessions is now 2 (../src/upf/context.c:252)
11/12 13:01:57.135: [gtp] INFO: gtp_connect() [192.168.7.180]:2152 (../lib/gtp/path.c:61)
11/12 13:01:57.170: [upf] INFO: [Added] Number of UPF-Sessions is now 3 (../src/upf/context.c:208)
11/12 13:01:57.171: [upf] INFO: UE F-SEID[UP:0xcd CP:0xf35] APN[ims] PDN-Type[1] IPv4[10.46.0.4] IPv6[] (../src/upf/context.c:498)
11/12 13:01:57.171: [upf] INFO: UE F-SEID[UP:0xcd CP:0xf35] APN[ims] PDN-Type[1] IPv4[10.46.0.4] IPv6[] (../src/upf/context.c:498)
11/12 13:03:18.034: [upf] INFO: [Removed] Number of UPF-sessions is now 2 (../src/upf/context.c:252)
11/12 13:03:18.751: [upf] INFO: [Added] Number of UPF-Sessions is now 3 (../src/upf/context.c:208)
```

Network Data Analytics Function (NWDAF)

In 5G architecture, NWDAF stands for Network Data Analytics Function—a pivotal component designed to bring intelligence and automation to the core network. Think of NWDAF as the “brain” that helps the network self-optimize, predict issues, and adapt dynamically.

NWDAF is a standardized analytics engine defined by 3GPP (TS 29.520) that:

- Collects data from various network functions (NFs), user equipment (UE), and OAM systems.
- Processes this data using AI/ML models.
- Exposes insights to other NFs for decision-making and automation

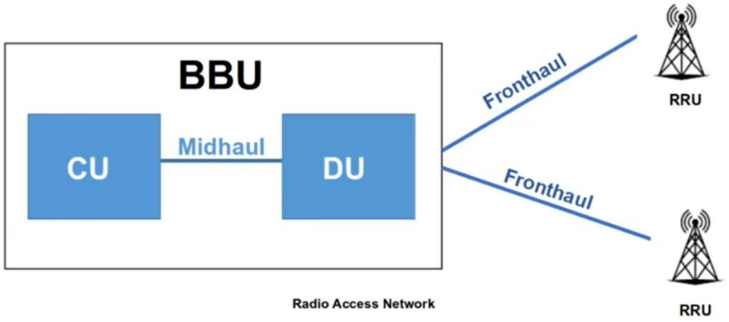
Objective: Check Service Status of NWDAF and its functionality

Procedure	Expected Output
Run command $ core-network status nwdaf	**Expected Output:** Service: coral5gs-nwdafd.service Active: active (running) Service ID: 1b87483a-305b-41e9-b670-2985996f8551 The service ID in the example provided is `*1b87483a-305b-41e9-b670-2985996f8551*`. Your specific service ID may vary depending on your system or setup.

5G Lab Book

Part -3: 5G NG Radio

System description -Integrated (BBU&RU) Architecture:



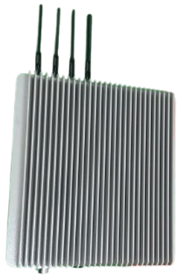
Components of Integrated Architecture:

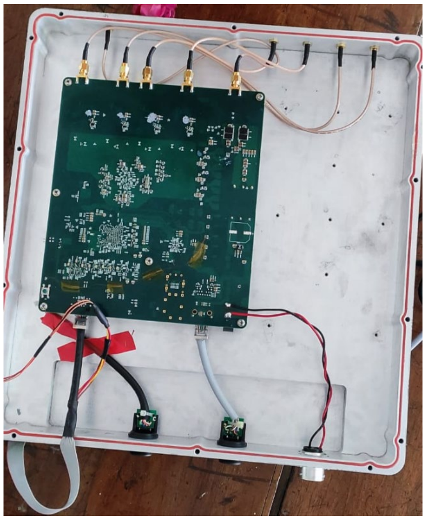
BBU (Baseband Unit): Handles signal processing, protocol stack, and communication with the core network.

RU (Radio Unit): Manages radio frequency (RF) functions, like sending/receiving wireless signals to/from UEs (User Equipment).

DU (Distributed Unit): Processes Layer 1 (PHY) and part of Layer 2 functions.

CU (Centralized Unit): Handles higher Layer 2 and Layer 3 functions, managing control-plane and user-plane traffic.





gNB of Resonous Technologies make (2T2R configuration) and its Inside view

To verify gNB support to Radio Resource Management functions

Introduction:

Key Radio Resource Management functions (RRM) Functions are

1. Radio Bearer Control

Technical Role:

- Manages the setup, modification, and release of **Data Radio Bearers (DRBs)** and **Signalling Radio Bearers (SRBs)**.
- Ensures QoS flows are mapped to appropriate bearers based on service requirements.

Key Messages/Indicators for Verification/Check:

- RRCConnectionSetup, RRCConnectionReconfiguration, and DRB-ToAddModList in NR-RRC.
- Bearer types: SRB1 (initial signalling), SRB2 (dedicated signalling), DRB (user data).

Outreach Angle:

Think of radio bearers as dedicated lanes on a highway—each one optimized for a specific type of traffic, whether it's voice, video, or control signals.

2. Radio Admission Control

Technical Role:

- Decides whether to accept or reject a new UE connection based on current resource availability.
- Prevents overload and ensures service continuity for existing users.

Key Messages/Indicators for Verification/Check:

- RRCConnectionRequest → establishmentCause
- RRCConnectionSetup → admission granted or denied
- gNB config: AdmissionControlEnabled = TRUE

Outreach Angle:

"Admission control is like a bouncer at a club—only letting in new guests if there's enough space to keep everyone comfortable."

3. Connection Mobility Control

Technical Role:

- Handles **intra-gNB** and **inter-gNB** handovers.
- Maintains seamless connectivity as UEs move across cells or coverage areas.

Key Messages/Indicators for Verification/Check:

- RRCReconfiguration with mobility parameters
- HandoverPreparation, HandoverRequest in NGAP
- targetCellId, handoverType, UEHistoryInformation

Outreach Angle:

"Mobility control ensures your video call doesn’t drop when you walk from one room to another—it hands you off smoothly between towers."

4. Dynamic Resource Allocation (UL/DL)

Technical Role:

- Allocates **Physical Resource Blocks (PRBs)** dynamically based on UE demand, QoS, and channel conditions.
- Operates via MAC scheduler in TDD/FDD modes.

Key Messages/Indicators for Verification/Check:

- UL-SCH and DL-SCH allocations
- BSR, SR, SchedulingRequestConfig
- MAC layer: ul-grant, dl-grant, CQI, MCS

To verify gNB support to Radio Bearer Control

To verify that a **gNB supports Radio Bearer Control** in a **5G Standalone (SA)** network, you'll want to validate its ability to establish, modify, and release **Data Radio Bearers (DRBs)** and **Signalling Radio Bearers (SRBs)** based on service requirements and QoS flows. Here's a detailed, step-by-step approach tailored for lab validation and outreach:

Radio Bearer Control is the gNB's function to manage the **radio interface bearers** that carry user data and signalling. In 5G SA, this includes:

- **SRB1/SRB2** for signalling
- **DRB** for user data
- Mapping of **QoS flows** to **DRBs** via **QoS Flow to DRB Mapping**

Step-by-Step Verification in Lab

Setup Environment

- Use a **UE simulator** or test device
- Connect to a **5G SA core** (AMF, SMF, UPF)
- Ensure gNB is configured for SA mode (no EN-DC fallback)

Trigger RRC Connection

- Power on UE and initiate registration
- Observe RRCConnectionRequest → RRCConnectionSetup → RRCConnectionSetupComplete

Key Messages/Indicators for Verification/Check:

- radioBearerConfig
- drb-ToAddModList
- srb-ToAddModList

Establish PDU Session

- UE sends NAS: Registration Request and PDU Session establishment Request
- SMF responds with session parameters
- gNB configures DRBs via RRCConnectionReconfiguration

Check for:

- DRB identity
- QoS Flow to DRB mapping
- SDAP and PDCP configurations

Capture and Analyze Traces

Use **Wireshark** with filters:

- nr-rrc
- ngap
- mac-lte

Display Fields to Annotate:

- nr-rrc.DRB-Identity
- nr-rrc.radioBearerConfig
- ngap.AMRBSetupList
- mac-lte.dl-grant, ul-grant

Modify or Release Bearers

- Trigger QoS change or session release
- Observe RRCConnectionReconfiguration for DRB modification or release

Confirmation:

- DRB release via drb-ToReleaseList Bearer reconfiguration with updated QoS

Set up steps:

- End to End systems to be configured with valid network parameters and the gNB is brought up. Attach UE with gNB.
- Establish a connection between the UE and gNB. Initiate video streaming to establish radio bearers. Modify QoS parameters for an existing bearer in Coral core and observe the changes. Release an existing radio bearer and verify its termination for Radio Bearer Control function.
- Dynamic allocation of resource to UE in both uplink and downlink (scheduling)
- Test 5G device from different location with respect to gNB to get variation in RSRP/RSRQ
- Measure the throughput using speed test application

- Check the PCAP logs of gNB for resource allocation

Test Method

1 To verify End to End systems are configured & gNB is radiating.
2.To verify successful bearer establishment, modification in RRC Connection Reconfiguration of Wireshark log.

Steps:

Filter nr-rrc/ RRC Reconfiguration
1.Verify SRB0, SRB 1& SRB 2 in RRC Setup and RRC Reconfiguration Message
2.Verify the PDU session establishment procedure for DRB 1 and DRB 2
3.Check the RRC Reconfiguration message in PCAP 7 show the mapping between QFI and DRB Identifier indicates Radio bearer control
4.Intiate flight mode on to initiate PDU session release followed by RRC Connection release.

Result:

SRB0, SRB1, SRB2, DRB1 bearer formed
rrcSetup
i)Item 2: id-SRBID ProtocolIE-Field id: id-SRBID (64) criticality: reject (0) value SRBID: 0
radioBearerConfig srb-ToAddModList: 1 item 0 SRB-ToAddMod srb-Identity: 1
iii)srb-ToAddModList
{
{
srb-Identity 2,
iv) drb-ToAddModList cnAssociation sdap-Config : pdu-Session 1, sdap-HeaderDL present, sdap-HeaderUL present, defaultDRB FALSE, mappedQoS-FlowsToAdd 1 drb-Identity 1,

Result:

Verify RRC Radio Bearers
Verified that gNB supports Radio Resource Management functions: Radio Bearer Control, Radio Admission Control, Connection Mobility Control, Dynamic allocation of resources to UEs in both uplink and downlink

To verify gNB support to Radio Admission Control

To verify that a **gNB supports Radio Admission Control** in a **5G Standalone (SA)** network, you'll need to confirm that it can evaluate incoming connection requests and decide whether to admit or reject them based on available radio resources. This function is critical for maintaining service quality and preventing network overload.

Radio Admission Control is the gNB's ability to:

- Assess **RRCConnectionRequest** messages from UEs.
- Decide whether to **accept or reject** based on resource availability, load, and policy.
- Ensure that only UEs with sufficient resources are admitted, preserving QoS for all.

1. **Setup Environment**
    - Use a **UE simulator** or multiple test UEs.
    - Connect to a **5G SA core** (AMF, SMF, UPF).
    - Configure gNB with **Admission Control enabled**:
        - AdmissionControlEnabled = TRUE
        - Set thresholds for PRB usage, UE count, QoS class.
2. **Trigger UE Attach**
    - Power on UE and initiate registration.
    - Observe signalling flow:
        - RRCConnectionRequest
        - RRCConnectionSetup
        - RRCConnectionSetupComplete

    Key Messages/Indicators for Verification/Check::
    - establishmentCause (e.g., emergency, mobile originated data)
    - UEIdentity
    - CellLoadInformation
3. **Simulate Resource Saturation**
    - Launch multiple UEs to exceed configured thresholds.
    - Monitor gNB behaviour:
        - Does it reject new connections?
        - Does it prioritize emergency or high-priority traffic?

Expected Behaviour:

- Admission rejected for low-priority UEs.
- Emergency UEs admitted even under load.

4. Capture and Analyze Traces

Use **Wireshark** with filters:

- nr-rrc
- ngap

Display Fields to Annotate:

- nr-rrc.establishmentCause
- nr-rrc.radioBearerConfig
- ngap.InitialUEMessage
- ngap.RRCSetupFailure (if rejection occurs)

5. Monitor KPIs and Logs

- Admission Success Rate
- RRC Setup Failure due to resource constraints
- gNB logs showing admission decisions

To verify gNB support to Connection Mobility Control

To verify that a **gNB supports Connection Mobility Control** in a **5G Standalone (SA)** network, you need to confirm its ability to manage seamless handovers and maintain UE connectivity as it moves across cells or gNBs. This function ensures uninterrupted service during mobility events—critical for real-time applications like voice, video, and IoT.

Connection Mobility Control is the gNB's capability to:

- Detect UE movement and trigger handover procedures.
- Coordinate with neighbouring cells or gNBs.
- Maintain session continuity across mobility events.

In 5G SA, this involves **intra-gNB**, **inter-gNB**, and **inter-PLMN** mobility using **RRC**, **NGAP**, and **XnAP**/F1AP protocols.

1. Setup Environment

- Use UE simulator with mobility profiles.
- Deploy multiple gNBs or configure multiple cells within a single gNB.
- Ensure handover parameters are enabled:
  - MobilityControlMode = HO
  - HOType = Intra/Inter

2. Trigger UE Movement

- Simulate UE moving from one cell to another.
- Observe signal strength thresholds crossing handover margins.

Expected Behaviour:

- gNB initiates handover preparation.
- Target cell/gNB accepts UE context.
- UE reconfigures and resumes data flow.

3. Capture and Analyze Signalling

Use **Wireshark** or protocol analyzer with filters:

- nr-rrc
- ngap
- xnap or f1ap (depending on architecture)

Key Messages/Indicators for Verification/Check:

Protocol	Message	Purpose
NR-RRC	RRCReconfiguration	Triggers handover with mobility parameters
NGAP	HandoverRequired, HandoverRequest, HandoverCommand	Coordinates handover between AMF and gNBs
XnAP/F1AP	HandoverPreparation, UEContextTransfer	Transfers UE context to target gNB

Important IEs:

- targetCellId
- handoverType
- UEHistoryInformation
- mobilityRestrictionList

4. Monitor KPIs and Logs

- Handover Success Rate
- Handover Preparation Time
- UE Context Transfer Success
- gNB logs showing mobility decisions and thresholds

To verify gNB support to Dynamic Allocation of Resources

To verify that a **gNB supports Dynamic Allocation of Resources** to UEs in both **uplink and downlink** in a **5G Standalone (SA)** network, you'll need to validate its real-time scheduling behaviour at the **MAC layer**, where Physical Resource Blocks (PRBs) are assigned based on UE demand, QoS, and radio conditions.

Dynamic Resource Allocation is the gNB's ability to:

- Monitor UE buffer status, channel quality, and QoS requirements.
- Dynamically assign PRBs for **UL-SCH** and **DL-SCH** transmissions.
- Adjust scheduling decisions in real time to optimize throughput and fairness.

Step-by-Step Verification in Lab

1. **Setup Environment**
    - Use a **UE simulator** or real test UEs.
    - Connect to a **5G SA core** (AMF, SMF, UPF).
    - Ensure gNB scheduler is configured for dynamic mode:
        - SchedulerType = Dynamic
        - CQIReportingEnabled = TRUE
2. **Trigger Data Sessions**
    - Initiate **PDU session establishment** for multiple UEs.
    - Start **high-throughput applications** (e.g., video streaming, file upload).
    - Observe uplink and downlink traffic patterns.

    Expected Behaviour:
    - gNB allocates PRBs dynamically based on traffic demand.
    - Scheduling decisions vary per TTI (Transmission Time Interval).
3. **Capture and Analyze MAC Layer Traces**

    Use **Wireshark** or protocol analyzer with filters:
    - mac-lte
    - nr-rrc

Key Display Fields to Annotate:

Layer	**Field**	**Description**
MAC	mac-lte.ul-grant	Uplink resource allocation
MAC	mac-lte.dl-grant	Downlink resource allocation
MAC	BSR, SR, CQI	Inputs to scheduler
RRC	SchedulingRequestConfig	UE request behaviour

4. Observe Scheduling Behavior

- Compare PRB allocation across UEs with different QoS profiles.
- Validate responsiveness to:
  - Buffer Status Reports (BSR)
  - Scheduling Requests (SR)
  - Channel Quality Indicators (CQI)

Indicators of Dynamic Allocation:

- PRB count varies per UE and per TTI.
- High-demand UEs receive more PRBs.
- CQI fluctuations impact MCS and PRB assignment.

5. Monitor KPIs and Logs

- PRB Utilization per UE
- Scheduler Latency
- Throughput per QoS Flow
- gNB logs showing scheduling decisions

To verify gNB support to Routing of User Plane data towards UPF(s)

To verify that **User Plane data is correctly routed toward UPF(s)** in a **5G Standalone (SA)** network, you need to validate the **data path establishment**, **tunnel configuration**, and **packet flow** between the **UE → gNB → UPF**. This ensures that user traffic (e.g., internet, video, VoLTE) is handled as per QoS and session parameters.

In 5G SA, the **User Plane Function (UPF)** handles actual user data, while control signalling is managed by AMF/SMF. The gNB routes user traffic to the correct UPF using **GTP-U tunnels** over **N3 interface**.

1. **Setup Environment**
   - UE simulator or test device
   - gNB configured for SA mode
   - 5G Core with AMF, SMF, UPF
   - Packet capture tool (e.g., Wireshark) on N3 interface
2. **Trigger PDU Session Establishment**
   - UE sends NAS: PDU Session Establishment Request
   - SMF selects UPF and responds with tunnel parameters
   - gNB configures DRBs and GTP-U tunnel

   Key Messages/Indicators for Verification/Check::
   - NGAP: InitialContextSetupRequest
   - NGAP: PDU Session Resource Setup Request
   - RRC: RRCConnectionReconfiguration
3. **Inspect Tunnel Configuration**

   Use Wireshark or gNB logs to verify:
   - **GTP-U Tunnel Setup**:
     - GTPv1-U packets on N3 interface
     - TEID (Tunnel Endpoint Identifier)
     - QoS Flow to DRB Mapping

Display Fields to Annotate:

- gtp.teid
- gtp.message_type == 255 (G-PDU)
- nr-rrc.DRB-Identity
- ngap.PDUSessionResourceSetupRequestTransfer

4. Send User Data

- UE initiates data session (e.g., ping, HTTP download)
- Capture packets on N3 interface
- Confirm GTP-U encapsulation and routing to UPF

Expected Behavior:

- GTP-U packets flow from gNB to UPF
- Correct TEID and QoS marking
- No packet loss or misrouting

5. Monitor KPIs and Logs

- Throughput per PDU session
- GTP-U packet count and latency
- UPF logs showing session context and routing

Test Set UP

- Set UP Steps End to End systems to be configured with valid network parameters and the gNB is brought up and attach UE with gNB.
- Register UE with gNodeB start the data transfer (Ping to 8.8.8.8) in UL and Verify GTP-U information from gNodeB PCAP log
- Verify the PDU session Resource setup Request for Tunnel ID.
- Verify the packet list to check the data being transferred using the Tunnel ID

Testing methodology:

UE received data and verified via iperf application in UE. To verify the GTP/UDP packets in gNB via Pcap logs

Steps:

To Login gNB via mobaxtrem,
Open mobaxtrem in laptop,
Type, root@ [gNB IP] PW: Run, tcpdump -I [Ethernet/optical interface of gNB] -w filename. pcap

Validation:

Open Wireshark app in laptop.
Open Wireshark > File > open pcap log > Apply UDP/GTP filters.

Result:

Tunnel establishment via PDU session resource
PDUSessionResourceSetupRequest
protocolIEs: 4 items
Item 0: id-AMF-UE-NGAP-ID

```
ProtocolIE-Field
id: id-AMF-UE-NGAP-ID (10)
criticality: reject (0)
value AMF-UE-NGAP-ID: 23
Item 1: id-RAN-UE-NGAP-ID
ProtocolIE-Field id: id-RAN-UE-NGAP-ID (85)
criticality: reject (0) value RAN-UE-NGAP-ID: 23

Item 2: id-
PDUSessionResourceSetupListSU
Req
ProtocolIE-Field
id: id-
PDUSessionResourceSetupListSU
Req (74)
criticality: reject (0)
value
PDUSessionResourceSetupListSU
Req: 1 item
Item 0
PDUSessionResourceSetupItemSU
Req
pDUSessionID: 1
pDUSessionNAS-PDU:
7e02a59878f3137e00680100552e0
109c211000901000631310101ff010
603f42403f4242905010a2d000522
04010306097900060120410101097
b001a8080210a0200000a8106c0a8
0f0a000d04c0a80f0a001002057825
0908696e7465726e65741201
Non-Access-Stratum 5GS
(NAS)PDU
Security protected NAS 5GS
message
Extended protocol discriminator:
5G mobility management messages
(126)
0000 .... = Spare Half Octet:0.... 0010 = Security
```

```
header type: Integrity protected and
ciphered (2)
Message authentication code:
0xa59878f3
Sequence number: 19
Encrypted data
s-NSSAI
sST: 01
sD: 030609
pDUSessionResourceSetupRequest
Transfer:
0000040082000a0c3b9aca00303b9
aca00008b000a01f0c0a80f0a0000d
2d700860001000088000700010000
091c00
PDUSessionResourceSetupRequest
Transfer
protocolIEs: 4 items
Item 0: id-
PDUSessionAggregateMaximumBi
tRate
ProtocolIE-Field
id: id-
PDUSessionAggregateMaximumBi
tRate (130)
criticality: reject (0)
value
PDUSessionAggregateMaximumBi
tRate
pDUSessionAggregateMaximumBit
RateDL: 1000000000bits/s
pDUSessionAggregateMaximumBit
RateUL: 1000000000bits/s
Item 1: id-UL-NGU-UPTNLInformation
ProtocolIE-Field
id: id-UL-NGU-UPTNLInformation
(139)
criticality: reject (0)
value
```

UPTransportLayerInformation:
gTPTunnel (0)
gTPTunnel
transportLayerAddress: c0a80f0a
[bit length 32, 1100 0000 1010
1000 0000 1111 0000 1010
decimal value 3232239370]
TransportLayerAddress (IPv4):
192.168.15.10
gTP-TEID: 0000d2d7

Result: Verified that gNB supports Routing of User Plane data towards UPF(s)

To verify gNB support to Routing of Control Plane information to AMF

To verify that **Control Plane information is correctly routed to the AMF** in a **5G Standalone (SA)** network, you need to confirm that the gNB and UE are properly exchanging **NAS signalling** via the **NGAP interface**, and that the AMF is receiving and responding to these messages as expected.

Control Plane Routing to AMF in 5G SA:

- The **gNB** handles RRC signalling with the UE.
- The **AMF** manages NAS signalling (e.g., registration, authentication, mobility).
- The **NGAP protocol** over the **N2 interface** connects gNB ↔ AMF for control signalling.

1. Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core with AMF, SMF, UPF
- Packet capture tool (e.g., Wireshark) on N2 interface

2. Trigger UE Registration

- UE sends RRCConnectionRequest → RRCConnectionSetup → RRCConnectionSetupComplete
- NAS message (Registration Request) is encapsulated in InitialUEMessage and forwarded to AMF

Expected Behavior:

- gNB forwards NAS messages to AMF via NGAP
- AMF responds with AuthenticationRequest, SecurityModeCommand, etc.

3. Capture and Analyse NGAP Signalling

Use **Wireshark** with filters:

- ngap
- nr-rrc

Key Messages/Indicators for Verification/Check:

Protocol	Message	Purpose
NGAP	InitialUEMessage	UE registration forwarded to AMF
NGAP	UplinkNASTransport	UE → AMF NAS messages
NGAP	DownlinkNASTransport	AMF → UE NAS messages
NGAP	NGSetupRequest/Response	gNB-AMF setup procedure

Important IEs:

- UE NGAP ID
- NAS-PDU
- AMF UE NGAP ID
- PLMN ID, TAI, S-NSSAI

4. Confirm AMF Response

- AMF sends back NAS messages via DownlinkNASTransport
- gNB delivers them to UE via RRC

Verify:

- Successful completion of registration and authentication
- Security context establishment
- PDU session setup initiation

5. Monitor KPIs and Logs

- NAS Message Success Rate
- Registration Success Rate
- AMF logs showing UE context creation and NAS handling

Outreach Tip

For public campaigns or training:

“Routing control messages in 5G is like sending instructions from a command center to field units—ensuring every device knows what to do, securely and instantly.”

Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core with AMF, SMF, UPF
- Packet capture tool (e.g., Wireshark) on **N2 interface**

Trigger UE Registration

- UE sends RRCConnectionRequest
- gNB responds with RRCConnectionSetup
- UE sends RRCConnectionSetupComplete with embedded NAS RegistrationRequest

Expected Behaviour:

- gNB encapsulates NAS message in NGAP: InitialUEMessage
- AMF receives and processes the NAS message

Capture and Analyse NGAP Signalling

Use **Wireshark** with filters:

- ngap
- nr-rrc

Key NGAP Messages/Indicators for Verification/Check:

Message	Purpose
NGSetupRequest/Response	Establishes gNB-AMF connection
InitialUEMessage	Carries UE's first NAS message to AMF
UplinkNASTransport	Sends NAS messages from UE to AMF
DownlinkNASTransport	Sends NAS messages from AMF to UE
NAS-PDU	Contains actual NAS message (e.g., RegistrationRequest)

Important IEs:

- UE NGAP ID
- AMF UE NGAP ID
- PLMN ID, TAI, S-NSSAI
- NAS-PDU (decoded to verify content)

Confirm AMF Response

- AMF sends back NAS messages (e.g., AuthenticationRequest, SecurityModeCommand)
- gNB delivers them to UE via RRCConnectionReconfiguration

Verify:

- Successful completion of registration and authentication
- Security context establishment
- PDU session setup initiation

Monitor KPIs and Logs

- NAS Message Success Rate
- Registration Success Rate
- AMF logs showing UE context creation and NAS handling
- gNB logs confirming NGAP message forwarding

Test Set Step:

- End to End systems to be configured with valid network parameters and the gNB is brought up.
- Configure AMF IP, N2 interface & ports in gNB & register UE with gNB.
- Verify using gNodeB PCAP logs for NGAP SETUP Request and Response

Testing methodology:

To verify that NGAP protocol message exchanges in Pcap log between gNB & AMF for UE PDU session,

Steps:

To Login gNB via mobaxtrem, Open mobaxtrem in laptop,
Type, root@ [gNB IP]
PW: root
Run, tcpdump -I [Ethernet/optical interface of gNB] -w filename. pcap

Validation:

Open Wireshark app in laptop.
Open Wireshark > File > open pcap log > Apply NGAP filters

Result:

NGAP messages shall updated successfully between gNB & AMF

NGSetupRequest
protocolIEs: 3 items

```
Item 0: id-
GlobalRANNodeID
ProtocolIE-Field
id: id-GlobalRANNodeID
(27)
criticality: reject (0)
value
GlobalRANNodeID:
globalGNB-ID (0)
globalGNB-ID
pLMNIdentity: 00f110
Mobile Country Code
(MCC): Unknown (001)
Mobile Network Code
(MNC): Unknown (01)
gNB-ID: gNB-ID (0)
gNB-ID: 0000000f [bit
length 32, 0000 0000
0000 0000 0000 0000
0000 1111 decimal value
15]
Item 1: id-
SupportedTAList
NG Application Protocol
```

(NGSetupResponse)

```
NGAP-PDU:
successfulOutcome (1)
successfulOutcome
```

Result : Verified that gNB supports Routing of Control Plane information towards AMF via pcap log.

To verify gNB support to Connection setup and release

To verify that a **gNB supports Connection Setup and Release** in a **5G Standalone (SA)** network, you need to validate its ability to establish and tear down **RRC connections** with the UE, and ensure proper signalling with the **AMF** via the **NGAP interface**. This is foundational to all UE interactions in the network.

- **Connection Setup**: Establishes the RRC connection between UE and gNB, enabling NAS signalling and PDU session setup.
- **Connection Release**: Terminates the RRC connection when no longer needed, freeing up radio resources.

1. Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core with AMF, SMF, UPF
- Packet capture tool (e.g., Wireshark) on N2 interface

2. Trigger RRC Connection Setup

- UE sends RRCConnectionRequest
- gNB responds with RRCConnectionSetup
- UE sends RRCConnectionSetupComplete with embedded NAS message

Expected Behavior:

- gNB assigns temporary UE ID
- RRC connection is established
- NAS signalling begins (e.g., RegistrationRequest to AMF)

3. Capture and Analyze Setup Signalling

Use **Wireshark** with filters:

- nr-rrc
- ngap

Key Messages/Indicators for Verification/Check:

Protocol	Message	Purpose
NR-RRC	RRCConnectionRequest	UE initiates connection
NR-RRC	RRCConnectionSetup	gNB configures radio resources
NR-RRC	RRCConnectionSetupComplete	UE confirms setup and sends NAS
NGAP	InitialUEMessage	gNB forwards NAS to AMF

Important IEs:

- establishmentCause
- UEIdentity
- PLMN ID, S-NSSAI

4. Trigger RRC Connection Release

- UE becomes idle or deregisters
- gNB sends RRCConnectionRelease to UE

Expected Behaviour:

- UE receives release command
- Radio resources are freed
- UE transitions to idle mode

5. Capture and Analyse Release Signalling

Key Messages/Indicators for Verification/Check:

- NR-RRC: RRCConnectionRelease

Check for:

- Release cause (e.g., normal release, inactivity)
- UE context removal in gNB and AMF

6. Monitor KPIs and Logs

- RRC Setup Success Rate
- RRC Release Success Rate
- gNB logs showing connection lifecycle
- AMF logs confirming UE context creation/removal

Test Set up:

- Run the gNodeB Connect the UE Call and Video streaming should happen
- Put UE in Airplane mode. This will disconnect the UE from gNodeB
- Remove from airplane mode Verify UE context release command & UE context release complete

Test Methodology:

UEContextReleaseCommand
protocolIEs: 2 items
Item 0: id-UE-NGAP-IDs
ProtocolIE-Field
id: id-UE-NGAP-IDs (114)
criticality: reject (0)
value
UE-NGAP-IDs: uE-NGAP-IDpair
(0) uE-NGAP-ID-pair
aMF-UE-NGAP-ID: 6
rAN-UE-NGAP-ID:
NGAP-PDU:
successfulOutcome (1)
successfulOutcome
procedureCode: id- UEContextRelease (41)
criticality: reject (0) value
UEContextReleaseComplete
protocolIEs: 3 items
Item 0: id-AMF-UE-NGAPID
ProtocolIE-Field
id: id-AMF-UE-NGAP-ID (10)
criticality: ignore (1)
value
AMF-UE-NGAP-ID: 6
Item 1: id-RAN-UE-NGAPID
ProtocolIE-Field
id: id-RAN-UE-NGAP-ID (85)
criticality: ignore (1)
value
RAN-UE-NGAP-ID: 6

Result: Verified that gNB supports Connection setup and release.

To verify gNB supports Scheduling & transmission of paging messages

To verify that a **gNB supports scheduling and transmission of paging messages** in a **5G Standalone (SA)** network, you need to confirm its ability to deliver paging notifications to idle-mode UEs based on core network triggers. Paging is essential for alerting UEs about incoming services like calls, data, or configuration updates.

Paging enables the network to **reach UEs in RRC_IDLE or RRC_INACTIVE states**. The AMF initiates paging via NGAP, and the gNB schedules and transmits the paging message over the air interface.

1. Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core with AMF
- Packet capture tool (e.g., Wireshark) on N2 interface
- Paging trigger mechanism (e.g., incoming call or data)

2. Trigger Paging Event

- Put UE in **RRC_IDLE** state
- Initiate a service that requires UE attention (e.g., MT call, downlink data)
- AMF sends NGAP: Paging message to gNB

Expected Behaviour:

- gNB receives paging request
- Schedules paging transmission in appropriate paging occasion
- UE receives and responds with RRCConnectionRequest

3. Capture and Analyse Paging Signalling

Use **Wireshark** with filters:

- ngap
- nr-rrc

Key Messages/Indicators for Verification/Check:

Protocol	Message	Purpose
NGAP	Paging	AMF → gNB paging request
NR-RRC	Paging	gNB → UE paging message
NR-RRC	RRCConnectionRequest	UE response to paging

Important IEs:

- UEIdentityIndexValue
- PagingDRX
- CNDomain
- PagingFrame, Paging Occasion

4. Confirm Scheduling Behaviour

- Check that gNB schedules paging in correct **paging frame and occasion**
- Validate that UE wakes up and responds within expected DRX cycle

Verify:

- Paging message transmitted on correct SSB
- UE response timing aligns with configured DRX

5. Monitor KPIs and Logs

- Paging Success Rate
- Paging Response Time
- gNB logs showing paging scheduling decisions
- AMF logs confirming UE reachability

Test Set Up

Verify End to End systems are configured & gNB is radiating. Register UE to gNobdeB and collect Wireshark log and check for paging message

Testing methodology:

Paging

```
protocolIEs: 2 items
Item 0: id-UEPagingIdentity
ProtocolIE-Field
id: id-UEPagingIdentity (115)
criticality: ignore (1)
value
UEPagingIdentity: fiveG-STMSI
(0)
fiveG-S-TMSI
aMFSetID: 0040 [bit length 10,
6 LSB pad bits, 0000 0000
01.. .... decimal value 1]
aMFPointer: 00 [bit length 6, 2
LSB pad bits, 0000 00..
decimal value 0]
fiveG-TMSI: 3221227364
(0xc0000764)
Item 1: id-TAIListForPaging
ProtocolIE-Field
id: id-TAIListForPaging (103)
criticality: ignore (1)
value
TAIListForPaging: 1 item
Item 0
TAIListForPagingItem
tAI
pLMNIdentity: 00f110
Mobile Country Code (MCC):
Unknown (001)
Mobile Network Code (MNC):
Unknown (01)
tAC: 1 (0x000001)
```

Result: Verified that gNB supports Scheduling and transmission of paging messages

To verify gNB supports Transport level packet marking in the uplink

To verify that a **gNB supports transport-level packet marking in the uplink** in a **5G Standalone (SA)** network, you need to confirm that it can apply **Differentiated Services Code Point (DSCP)** or other QoS-related markings to **uplink IP packets** before forwarding them to the **User Plane Function (UPF)** over the **N3 interface**. This is essential for enabling QoS enforcement across the transport network.

In 5G SA **Transport-Level Packet Marking**, **Uplink packets** from UE are encapsulated in **GTP-U** by the gNB. The gNB can apply **DSCP markings** to the outer IP header of GTP-U packets. These markings help routers and switches prioritize traffic based on QoS class.

1. Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core with UPF
- Packet capture tool (e.g., Wireshark) on **N3 interface**
- Transport network with DSCP-aware routers (optional)

2. Trigger Uplink Traffic

- Establish PDU session with specific QoS Flow (e.g., 5QI = 1 for conversational voice)
- UE sends uplink data (e.g., VoIP, ping, video upload)

Expected Behaviour:

- gNB encapsulates UE data in GTP-U
- Outer IP header of GTP-U packet includes DSCP marking

3. Capture and Analyse Packets

Use **Wireshark** on N3 interface with filters:

- ip.dsfield (to inspect DSCP value)
- gtp (to isolate GTP-U packets)

Key Messages/Indicators for Verification/Check:

Field	Description
ip.dsfield	DSCP value in outer IP header
gtp.teid	Tunnel Endpoint Identifier
gtp.message_type == 255	G-PDU (user data)
ip.src, ip.dst	gNB ↔ UPF IP addresses

Verify:

- DSCP value matches expected marking for the QoS Flow
- Consistency across multiple packets and sessions

4. Cross-Check with QoS Configuration

- Inspect gNB configuration:
  - Mapping of **5QI → DSCP**
  - Transport profile settings
- Confirm that DSCP marking is enabled for uplink traffic

5. Monitor KPIs and Logs

- DSCP marking success rate (if supported by gNB logs)
- QoS enforcement logs in UPF or transport routers
- Packet drop or delay metrics for marked vs. unmarked traffic

Test Set Up:

verify End to End systems are configured & gNB is radiating

- End to End systems to be configured with valid network parameters and the gNB is brought up and connected with Sngrep IMS node& attach UE.
- In Wireshark capture apply filters ngap || gtp ,
- Verify TEID match in the PDUsessionsetuipRequest and GTP packets

Testing methodology:

PDUSessionResourceSetupRequest

```
protocolIEs: 4 items
Item 0: id-AMF-UENGAP-ID
ProtocolIE-Field
id: id-AMF-UE-NGAPID (10)
criticality: reject (0)
value
AMF-UE-NGAP-ID: 23
Item 1: id-RAN-UENGAP-ID
ProtocolIE-Field
id: id-RAN-UE-NGAPID (85)
criticality: reject (0)
value
RAN-UE-NGAP-ID: 23
Item 2: id-PDUSessionResourceSetupListSUReq
ProtocolIE-Field
id: id-PDUSessionResourceSetupListSUReq (74)
criticality: reject (0)
value
PDUSessionResourceSetupListSUReq: 1 item
Item 0
PDUSessionResourceSetupItemSUReq
pDUSessionID: 1
pDUSessionNAS-PDU:
7e02a59878f3137e0068
0100552e0109c2110009
01000631310101ff0106
03f42403f4242905010a
2d000522040103060979
00060120410101097b00
1a8080210a0200000a81
06c0a80f0a000d04c0a8
```

```
0f0a0010020578250908
696e7465726e65741201
Non-Access-Stratum
5GS (NAS)PDU
Security protected NAS
5GS message
Extended protocol
discriminator: 5G
mobility management
messages (126)
0000 .... = Spare Half
Octet:
0
.... 0010 =
Security header type:
Integrity protected and
ciphered (2)
Message authentication
code: 0xa59878f3
Sequence number: 19
Encrypted data
s-NSSAI
sST: 01
sD: 030609
UPTransportLayerInformation: gTPTunnel (0)
gTPTunnel
transportLayerAddress:
c0a80f0a [bit length 32,
1100 0000 1010 1000
0000 1111 0000 1010
decimal value
3232239370]
TransportLayerAddress
(IPv4): 192.168.15.10
gTP-TEID:
0x00009f71.
Internet Protocol Version
4, Src: 192.168.15.15,
Dst: 192.168.15.10
0100 .... = Version: 4
.... 0101 = Header
Length: 20 bytes (5)
Differentiated
```

Services Field: 0x00
(DSCP: CS0, ECN: Not
ECT)
0000 00.. =
**Differentiated Services
Codepoint:** Default (0)
.... ..00 = Explicit
Congestion Notification:
Not ECN-Capable
Transport (0)
**GPRS Tunnelling
Protocol**
Flags: 0x34
001. .... = Version:
GTP release 99 version
(1)
...1 .... = Protocol
type: GTP (1)
.... 0... = Reserved:0.... .1.. = Is Next
Extension Header
present?: Yes
.... ..0. = Is
Sequence Number
present?: No
....0 = Is N-PDU
number present?: No
Message Type: TPDU
(0xff)
Length: 72
TEID: 0x00009f71

Result: Verified that gNB supports Transport level packet marking in the uplink

To verify gNB supports Session Management in 5G Standalone (SA)

To verify that a **gNB supports Session Management in 5G Standalone (SA)** mode, you'll want to check for its ability to handle **PDU session establishment** and interact correctly with the **AMF and SMF** over the N2 and N11 interfaces. Here's how you can validate it:

Key Indicators of Session Management Support in gNB

1. RRC Reconfiguration Message

- **Purpose**: Triggers PDU session setup
- **Direction**: gNB → UE
- **Includes**: SessionResourceSetupList, QoS parameters, and tunnel info
- **Verification**: Confirm presence of sessionResourceSetupList in Wireshark or trace logs

2. NGAP Signalling: Session Resource Setup

- **Key Messages/Indicators for Verification/Check:** NGAP: InitialContextSetupRequest or SessionResourceSetupRequest
- **Direction**: AMF → gNB
- **Content**: Contains session parameters from SMF
- **Followed by**: SessionResourceSetupResponse from gNB → AMF

3. Support for N2 Interface

- gNB must support **NGAP protocol** over N2 to communicate session setup with AMF

4. Capability Declaration

- During NG Setup Request, gNB declares supported procedures
- Look for Supported Message List including:
  - SessionResourceSetup
  - SessionResourceModify
  - SessionResourceRelease

Wireshark Display Filters for Validation

```
ngap.SessionResourceSetupRequest
ngap.SessionResourceSetupResponse
rrc.reconfiguration
```

A **PDU (Protocol Data Unit) session** is the logical connection that enables a UE (User Equipment) to send and receive data over the 5G Core network. It's essential for accessing services like internet, VoNR, or enterprise slices.

Step-by-Step PDU Session Establishment Flow

1. UE Sends Registration Request

- NAS message: Registration Request
- Sent via InitialUEMessage (NGAP) from gNB → AMF
- Includes UE capabilities and request for PDU session

2. Authentication & Security Setup

- AMF coordinates with AUSF/UDM for UE authentication
- Security Mode Command/Complete exchanged to secure NAS signalling

3. UE Sends PDU Session Establishment Request

- NAS message: PDU Session Establishment Request
- Includes:
    - Requested DNN (Data Network Name)
    - S-NSSAI (Slice info)
    - Session type (IPv4/IPv6)
    - SSC mode

4. AMF Forwards Request to SMF

- AMF → SMF via N11 interface
- SMF selects appropriate UPF and allocates tunnel parameters

5. SMF Sends Session Setup Info to AMF

- Includes:
    - QoS parameters
    - Tunnel endpoint info (TEID, IP)
    - UL CL (if applicable)

6. AMF Sends UEContextSetupRequest to gNB

- NGAP message
- Contains SessionResourceSetupList with tunnel and QoS info

7. gNB Sends RRC Reconfiguration to UE

- RRC message: RRC Reconfiguration
- Activates DRBs and configures QoS flows

8. UE Responds with RRC Reconfiguration Complete

- Confirms setup of radio bearers

9. gNB Sends UEContextSetupResponse to AMF

- Confirms successful setup of session resources

10. AMF Sends PDU Session Establishment Accept to UE

- NAS message
- UE can now start data transfer

Wireshark Display Filters for Validation: Key Messages/Indicators for Verification/Check:

plaintext
ngap.InitialUEMessage
ngap.UEContextSetupRequest
ngap.UEContextSetupResponse
nas_5gs.pdu_session_establishment_request
nas_5gs.pdu_session_establishment_accept
rrc.reconfiguration

Test Set Up:

verify End to End systems are configured & gNB is radiating.

- End to End systems to be configured with valid network parameters and the gNB is brought up.
- Register the UE with NGC.
- In Wireshark log filter out for NGAP message
- Verify PDUsession setup request and PDU session response message alternatively ,check for PDU session Establishment Request and PDU session Establishment Accept message in Wireshark log

Testing methodology:

From gNodeB PCAP logs:
Verify PDU session setup
request and PDU session
response message
PDUSessionResourceSetupRequest
protocolIEs: 3 items
Item 0: id-AMF-UE-NGAP-ID
ProtocolIE-Fieldid: id-AMF-UE-NGAP-ID (10)
criticality: reject (0)
value AMF-UE-NGAP-ID: 23
Item 1: id-RAN-UE-NGAP-ID

```
ProtocolIE-Field
id: id-RAN-UE-NGAP-ID
(85) criticality: reject (0)
value
RAN-UE-NGAP-ID: 23
Item 2: id-
PDUSessionResourceSetupList
SUReq
ProtocolIE-Field id: id-
PDUSessionResourceSetupList
SUReq (74)
criticality: reject (0)
value
PDUSessionResourceSetupList
SUReq: 1 item
Item 0
PDUSessionResourceSetupIte
mSUReq
pDUSessionID: 2
pDUSessionNAS-PDU:
7e02766ff46a147e0068010059
2e020ac211000901000631310
101ff0106010f0a0105fa59322
905010a2e0005220401030609
7900060120410101057b00218
080210a0200000a8106c0a80f0
a000d04c0a80f0a000c04c0a80
f0a0010020578250403696d73
1202
Non-Access-Stratum 5GS
(NAS)PDU
Security protected NAS 5GS
message
Extended protocol
discriminator: 5G mobility
management messages (126)
0000 .... = Spare Half Octet: 0.... 0010 = Security
header type: Integrity protected
and ciphered (2)
Message authentication code:
0x766ff46a
Sequence number: 20
Encrypted data
```

s-NSSAI
sST: 01
sD: 030609
NG Application Protocol
(PDUSessionResourceSetupResponse)
NGAP-PDU: successfulOutcome (1) successfulOutcome

Result: Verified that gNB supports Session Management

To verify gNB support to QoS Flow management & mapping to data radio bearers

To verify that a **gNB supports QoS Flow Management and Mapping to Data Radio Bearers (DRBs)** in a **5G Standalone (SA)** network, you need to confirm its ability to correctly interpret QoS parameters from the core network and configure the radio interface accordingly. This ensures that each **QoS Flow** is properly mapped to a **DRB**, enabling differentiated treatment of user traffic.

QoS Flow Management and DRB Mapping in 5G SA:

- The **SMF** defines QoS Flows with parameters like **5QI**, **ARP**, and **GBR**.
- The **gNB** maps these QoS Flows to **DRBs** using **SDAP** and **PDCP** configurations.
- Multiple QoS Flows can be multiplexed over a single DRB, or separated based on policy.

1. Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core: AMF, SMF, UPF
- Packet capture tool (e.g., Wireshark) on N2 and N3 interfaces

2. Trigger PDU Session with Multiple QoS Flows

- UE sends NAS: PDU Session Establishment Request
- SMF responds with multiple QoS Flows (e.g., 5QI = 1, 9, 65)
- gNB receives NGAP: PDU Session Resource Setup Request with QoS Flow parameters

  Expected Behaviour:
- gNB configures DRBs and maps QoS Flows accordingly
- UE receives RRCConnectionReconfiguration with SDAP and DRB setup

3. Capture and Analyse Signalling

Use **Wireshark** with filters:

- ngap
- nr-rrc

Key Messages to Inspect:

Protocol	Message	Purpose
NGAP	PDU Session Resource Setup Request	SMF → gNB QoS Flow info
NR-RRC	RRCConnectionReconfiguration	DRB and SDAP setup
NGAP	PDU Session Resource Setup Response	gNB → AMF confirmation

Important Information Elements:

- QoS Flow Identifier
- QoS Parameters (5QI, GBR, ARP)
- QoS Flow to DRB Mapping
- SDAP Configuration (default/reflective mapping)

4. Validate Mapping Behaviour

- Confirm that each QoS Flow is mapped to the correct DRB
- Check SDAP settings: sdap-Config, defaultDRB, reflective-QoS
- Observe packet treatment (e.g., priority, latency) across flows

Verify:

- Mapping consistency with SMF policy
- DRB configuration matches QoS expectations
- No misrouting or QoS violations

5. Monitor KPIs and Logs

- QoS Flow Setup Success Rate
- DRB Mapping Accuracy

gNB logs showing SDAP and DRB configuration

Test Set Up:

- End to End systems to be configured with valid network parameters and the gNB is brought up and attach UE with gNB.
- In the Wireshark log check the following:
- Apply ngap filter and Check PDUsessionResourceSetuprequest
- Apply the filter for nrrc message and check for RRC reconfiguration during Registration Accept
- In the RRC Reconfiguration message check the QFI/5QI That was created during PDUsessionResourceSetuprequest is correctly assigned a Data Radio Bearer.

Test Methodology

PDU Session Resource setup
Request for Radio bearer control.gnb/core network assign QFI:1 for internet default bearer with QFI value of 9
(**PDUSessionResourceSetup)**

```
NGAP-PDU:
initiatingMessage (0)
initiatingMessage
procedureCode: id-
PDUSessionResourceSetup
(29)
PDUSessionResourceSetupList
SUReq
ProtocolIE-Field
id: id- PDUSessionResourceSetupList
SUReq (74)
value PDUSessionResourceSetupList
SUReq: 1 item
Item 0
PDUSessionResourceSetupIte
mSUReq
pDUSessionID: 2
Non-Access-Stratum 5GS
(NAS)PDU
Security protected NAS 5GS
message
Extended protocol
discriminator: 5G mobility
management messages (126)
0000 .... = Spare Half Octet: 0.... 0010 = Security
header type: Integrity protected
and ciphered (2)
Message authentication code:
0x766ff46a
Sequence number: 20
Encrypted data
s-NSSAI
sST: 01
sD: 030609
value
PDUSessionAggregateMaximu
mBitRate
```

```
pDUSessionAggregateMaximu
mBitRateDL: 3850000bits/s
pDUSessionAggregateMaximu
mBitRateUL: 1530000bits/s
QosFlowSetupRequestList: 1
item
Item 0
QosFlowSetupRequestItem
qosFlowIdentifier: 1
qosFlowLevelQosParameters
qosCharacteristics:
nonDynamic5QI (0)
nonDynamic5Q
fiveQI: 9
```

Result: Verified that gNB supports QoS Flow management and mapping to data radio bearers.

To verify gNB support to paging functionality in 5G SA network

To find Paging in the network

Test Set Up steps:
verify End to End systems are configured & gNB is radiating

- End to End systems to be configured with valid network parameters and the gNB is brought up.
- Register the UE with NGC from flight mode.
- Keep the UE is in idle state & initiate MT call to the UE.

Testing methodology:
To verify that UE registration procedure is successful through UE context creation between gNB & AMF

Result:
To verify the paging message initiated from NGC to gNB to UE are successful in Wireshark log .

Steps:

Open mobaxtrem in laptop,
Type, root@ [gNB IP]
Run, tcpdump -I [Ethernet/optical interface of gNB] -w filename. pcap

Validation:

Open Wireshark → File → open pcap log → Apply NGAP

Result:
Paging shall be successful
between UE & gNB

Result: Verified the paging message.

To verify **paging functionality** in a **5G Standalone (SA)** network, you need to confirm that the gNB correctly receives paging requests from the AMF and schedules transmission of paging messages to UEs in **RRC_IDLE** or **RRC_INACTIVE** states. Paging is essential for alerting UEs about incoming services like calls, data, or configuration updates.

Here's a detailed, lab-ready guide tailored for your validation and outreach efforts:

What Is Paging in 5G SA?

Paging allows the network to reach UEs that are not actively connected (i.e., in idle or inactive mode). The **AMF** initiates paging via **NGAP**, and the **gNB** schedules and transmits the paging message over the air interface.

Step-by-Step Verification in Lab

1. Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core: AMF, SMF
- Packet capture tool (e.g., Wireshark) on **N2 interface**
- Paging trigger mechanism (e.g., incoming call, downlink data)

2. Trigger Paging Event

- Put UE in **RRC_IDLE** or **RRC_INACTIVE** state
- Initiate a service that requires UE attention (e.g., MT call, downlink packet)
- AMF sends NGAP: Paging message to gNB

Expected Behavior:

- gNB receives paging request
- Schedules paging transmission in correct **Paging Occasion**
- UE wakes up and responds with RRCConnectionRequest

3. Capture and Analyze Paging Signalling

Use **Wireshark** with filters:

- ngap
- nr-rrc

Key Messages to Inspect:

Protocol	**Message**	**Purpose**
NGAP	Paging	AMF → gNB paging request
NR-RRC	Paging	gNB → UE paging message
NR-RRC	RRCConnectionRequest	UE response to paging

Important IEs:

- UEIdentityIndexValue
- CNDomain
- PagingDRX
- PagingFrame, Paging Occasion

4. Confirm Scheduling Behavior

- Verify that gNB transmits paging in the correct **Paging Frame** and **Paging Occasion**
- Confirm UE response timing aligns with configured **DRX cycle**

5. Monitor KPIs and Logs

- Paging Success Rate
- Paging Response Time
- gNB logs showing paging scheduling decisions
- AMF logs confirming UE reachability

To verify a Successful Registration of UE

To verify that the **UE registration procedure is successful** via **UE context creation between gNB and AMF** in a 5G Standalone (SA) network, you can follow this structured approach using signalling trace analysis (e.g., Wireshark):

Key Steps to Verify UE Registration Success

1. Initial UE Message

- **Message**: InitialUEMessage
- **Protocol**: NGAP
- **Direction**: gNB → AMF
- **Content**: Contains NAS Registration Request from UE

2. Authentication & Security Procedures

- **Messages**:
  - Authentication Request / Response
  - Security Mode Command / Complete
- **Protocol**: NAS (encapsulated in NGAP)
- **Direction**: AMF ↔ UE (via gNB)

3. UE Context Setup

- **Message**: UEContextSetupRequest
- **Protocol**: NGAP
- **Direction**: AMF → gNB
- **Purpose**: Establish UE context at gNB with security and bearer parameters

4. UE Context Setup Response

- **Message**: UEContextSetupResponse
- **Protocol**: NGAP
- **Direction**: gNB → AMF
- **Confirmation**: Indicates successful context setup

5. Registration Accept

- **Message**: NAS Registration Accept
- **Direction**: AMF → UE (via gNB)
- **Significance**: Confirms UE is registered in 5G network

6. Registration Complete

- **Message**: NAS Registration Complete
- **Direction**: UE → AMF (via gNB)
- **Final Confirmation**: UE acknowledges successful registration

Wireshark Display Filters (NGAP Layer)

Use these filters to isolate relevant messages:

plaintext

```
ngap.InitialUEMessage
ngap.UEContextSetupRequest
ngap.UEContextSetupResponse
```

And for NAS messages within NGAP:

plaintext

```
nas_eps.registration_request
nas_eps.registration_accept
nas_eps.registration_complete
```

A Sample Trace Resource

You can download Dataset1.pcapng from this GitHub repository: Western-OC2-Lab/5G-Core-Networks-Datasets

Highlights of the trace:

- Captures **initial UE registration** procedure
- Includes **NGAP and NAS signalling**
- Shows interactions across 5GC functions like AMF, AUSF, UDM, etc.
- Compatible with **Wireshark** for protocol-level inspection

Suggested Display Filters for Analysis

To isolate key messages in Wireshark:

plaintext

```
ngap.InitialUEMessage
ngap.UEContextSetupRequest
ngap.UEContextSetupResponse
nas_5gs.registration_request
nas_5gs.registration_accept
nas_5gs.registration_complete
```

To find PDU Session Working

To determine whether a **PDU Session is working in a 5G Standalone (SA) network**, you'll want to trace the signalling flow from UE registration to session establishment and monitor key indicators across NAS and NGAP layers. Here's a structured approach tailored for your validation and outreach goals:

Step-by-Step Guide to Identify a Working PDU Session

1. **Start with UE Registration**
   - Confirm successful completion of RRC and NAS registration.
   - Look for Registration Accept and Registration Complete messages.
   - 5G SA Cell Search & Network Entry Matrix: 5G Interview Topics explains how initial access and cell search set the stage for session setup.

2. **Check NAS Transport for PDU Session Request**
   - UE sends PDU Session Establishment Request via UL NAS Transport.
   - Validate presence of parameters like SSC Mode, DNN, S-NSSAI, and PDU Session ID.
   - 5G Session Management Signalling Decoded: In-Depth ... breaks down these NAS messages and their significance.

3. **Monitor SMF and UPF Interaction**
   - SMF selects appropriate UPF and allocates tunnel endpoints.
   - Look for PDU Session Establishment Accept in DL NAS Transport.
   - Packet Core - 5G Standalone (3) - Create PDU Session Call ... visualizes this call flow with annotated protocol stack layers.

4. **Verify Bearer Setup via NGAP**
   - NGAP messages like Initial Context Setup Request and PDU Session Resource Setup Request confirm bearer creation.
   - Check for successful PDU Session Resource Setup Response.

5. **Confirm Data Path Activation**
    - Use Wireshark to inspect GTP-U tunnels between gNB and UPF.
    - Look for user-plane traffic with matching TEIDs and QoS Flow Identifiers.
    - 5G Registration and PDU Session using Pangolin | 5G | 5G ... demonstrates validation using Pangolin tool.

6. **Cross-check with Core Architecture**
    - Ensure alignment with network slicing, QoS enforcement, and session continuity.
    - What is 5G Core Network Architecture? Take a Look With ... provides a holistic view of how PDU sessions fit into the broader 5GC design.

7. **Training & Outreach Support**
   For public-facing or training materials, PDU Session Establishment - Part of 5G Course (Link in ... offers a simplified walkthrough ideal for workshops or campaign content

To find PDU Session working:

Test Set Up steps:
verify End to End systems are configured & gNB is radiating

- End to End systems to be configured with valid network parameters and the gNB is brought up &.
- Register the UE with NGC and SNGREP IMS node.
- Perform VoNR call from UE.

- Deregister the UE from the NGC.

Testing methodology:

To verify that PDU session in RRC connection Reconfiguration in PDU session establishment request/accept in pcap log.

The UE shall successfully update the pdu session modification message in Wireshark log.

Result: Verified that transport network layer is built on IP transport and GTP-U is used on top of UDP/IP to carry the user plane PDUs between the NG-RAN node and the UPF.

To verify that a PDU Session is working correctly in a 5G Standalone (SA) network, you need to confirm successful signalling between the UE, gNB, and core network (AMF, SMF, UPF), and validate that user data is flowing through the established session. Here's a structured, lab-ready approach tailored for your technical validation and outreach needs:

What Is a PDU Session?

A PDU (Protocol Data Unit) Session is the logical connection that enables user data (e.g., internet, voice, video) to flow between the UE and the UPF. It includes:

- Session establishment signalling
- QoS flow configuration
- DRB setup at the gNB
- GTP-U tunnel creation to UPF

Step-by-Step Verification in Lab

1. Setup Environment

- UE simulator or test device
- gNB configured for SA mode
- 5G Core: AMF, SMF, UPF
- Packet capture tool (e.g., Wireshark) on N2 and N3 interfaces

2. Trigger PDU Session Establishment

- After registration, UE sends: NAS: PDU Session Establishment Request
- AMF forwards to SMF
- SMF selects UPF and returns tunnel and QoS parameters
- gNB receives: NGAP: PDU Session Resource Setup Request
- gNB configures DRBs and sends: RRCConnectionReconfiguration to UE

Expected Behaviour:

- UE receives session config
- DRBs and QoS flows are mapped
- GTP-U tunnel is established to UPF

3. Capture and Analyze Signalling

Use Wireshark with filters:

- ngap
- nr-rrc
- gtp

Key Messages to Inspect:

Protocol	Message	Purpose
NAS	PDU Session Establishment Request	UE initiates session
NGAP	PDU Session Resource Setup Request	SMF → gNB session config
NR-RRC	RRCConnectionReconfiguration	DRB and SDAP setup
GTP-U	G-PDU	User data encapsulated for UPF

Important IEs:

- PDU Session ID
- QoS Flow Identifier
- DRB Identity
- TEID (Tunnel Endpoint Identifier)
- SDAP Configuration

4. Send and Monitor User Data

- UE initiates data transfer (e.g., ping, HTTP)
- Capture GTP-U packets on N3 interface
- Confirm DSCP marking (if applicable) and packet routing

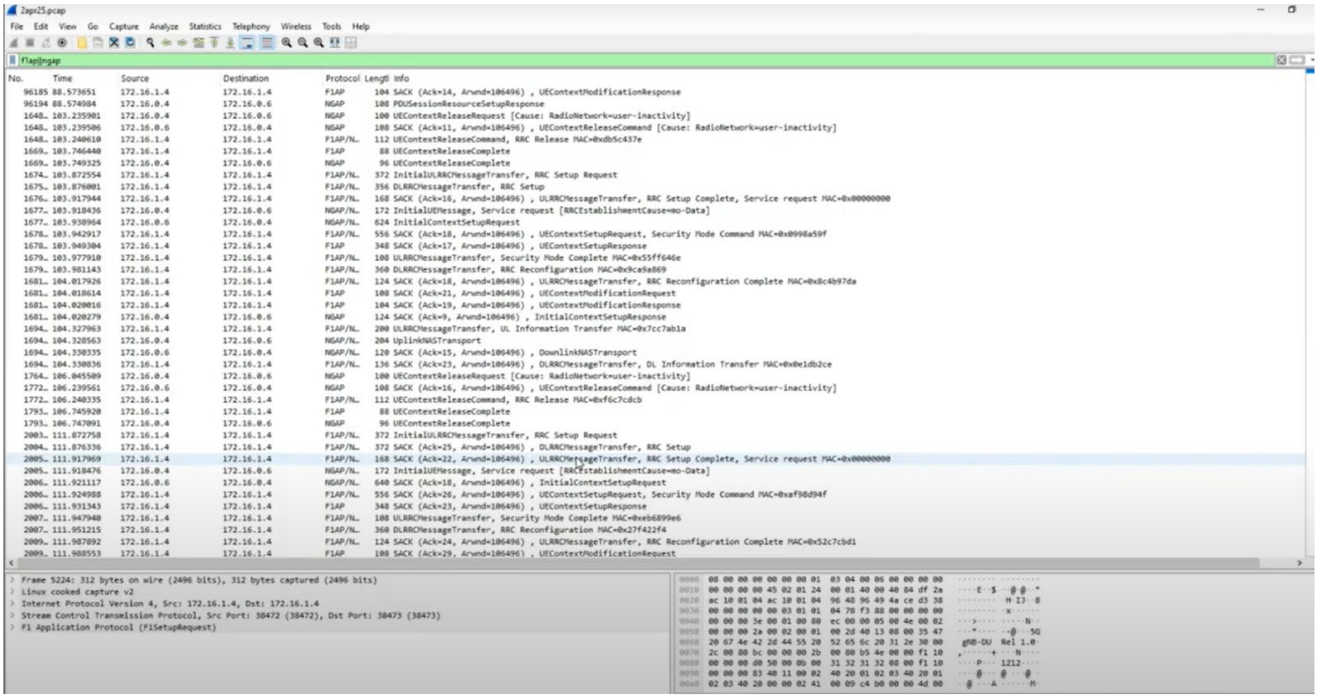
Verify:

- GTP-U packets flow from gNB to UPF
- Correct TEID and QoS mapping
- No packet loss or misrouting

5. Monitor KPIs and Logs

- PDU Session Setup Success Rate
- DRB Establishment Success
- QoS Flow Mapping Accuracy

- Throughput and latency metrics
- gNB and SMF logs showing session lifecycle


2apr25.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
f1ap||ngap

```
No.      Time        Source       Destination   Protocol  Lengtl Info
  96185 88.573651   172.16.1.4   172.16.1.4    F1AP        104 SACK (Ack=14, Arwnd=106496) , UEContextModificationResponse
  96194 88.574984   172.16.0.4   172.16.0.6    NGAP        108 PDUSessionResourceSetupResponse
  1648… 103.235901  172.16.0.4   172.16.0.6    NGAP        100 UEContextReleaseRequest [Cause: RadioNetwork=user-inactivity]
  1648… 103.239506  172.16.0.6   172.16.0.4    NGAP        108 SACK (Ack=11, Arwnd=106496) , UEContextReleaseCommand [Cause: RadioNetwork=user-inactivity]
  1648… 103.240610  172.16.1.4   172.16.1.4    F1AP/N…     112 UEContextReleaseCommand, RRC Release MAC=0xdb5c437e
  1669… 103.746440  172.16.1.4   172.16.1.4    F1AP         88 UEContextReleaseComplete
  1669… 103.749325  172.16.0.4   172.16.0.6    NGAP         96 UEContextReleaseComplete
  1674… 103.872554  172.16.1.4   172.16.1.4    F1AP/N…     372 InitialULRRCMessageTransfer, RRC Setup Request
  1675… 103.876001  172.16.1.4   172.16.1.4    F1AP/N…     356 DLRRCMessageTransfer, RRC Setup
  1676… 103.917944  172.16.1.4   172.16.1.4    F1AP/N…     168 SACK (Ack=16, Arwnd=106496) , ULRRCMessageTransfer, RRC Setup Complete, Service request MAC=0x00000000
  1677… 103.918436  172.16.0.4   172.16.0.6    NGAP/N…     172 InitialUEMessage, Service request [RRCEstablishmentCause=mo-Data]
  1677… 103.938964  172.16.0.6   172.16.0.4    NGAP/N…     624 InitialContextSetupRequest
  1678… 103.942917  172.16.1.4   172.16.1.4    F1AP/N…     556 SACK (Ack=18, Arwnd=106496) , UEContextSetupRequest, Security Mode Command MAC=0x0998a59f
  1678… 103.949304  172.16.1.4   172.16.1.4    F1AP        348 SACK (Ack=17, Arwnd=106496) , UEContextSetupResponse
  1679… 103.977910  172.16.1.4   172.16.1.4    F1AP/N…     108 ULRRCMessageTransfer, Security Mode Complete MAC=0x55ff646e
  1679… 103.981143  172.16.1.4   172.16.1.4    F1AP/N…     360 DLRRCMessageTransfer, RRC Reconfiguration MAC=0x9ca9a869
  1681… 104.017926  172.16.1.4   172.16.1.4    F1AP/N…     124 SACK (Ack=18, Arwnd=106496) , ULRRCMessageTransfer, RRC Reconfiguration Complete MAC=0x8c4b97da
  1681… 104.018614  172.16.1.4   172.16.1.4    F1AP        108 SACK (Ack=21, Arwnd=106496) , UEContextModificationRequest
  1681… 104.020016  172.16.1.4   172.16.1.4    F1AP        104 SACK (Ack=19, Arwnd=106496) , UEContextModificationResponse
  1681… 104.020279  172.16.0.4   172.16.0.6    NGAP        124 SACK (Ack=9, Arwnd=106496) , InitialContextSetupResponse
  1694… 104.327963  172.16.1.4   172.16.1.4    F1AP/N…     200 ULRRCMessageTransfer, UL Information Transfer MAC=0x7cc7ab1a
  1694… 104.328563  172.16.0.4   172.16.0.6    NGAP/N…     204 UplinkNASTransport
  1694… 104.330335  172.16.0.6   172.16.0.4    NGAP/N…     120 SACK (Ack=15, Arwnd=106496) , DownlinkNASTransport
  1694… 104.330836  172.16.1.4   172.16.1.4    F1AP/N…     136 SACK (Ack=23, Arwnd=106496) , DLRRCMessageTransfer, DL Information Transfer MAC=0x0e1db2ce
  1764… 106.045509  172.16.0.4   172.16.0.6    NGAP        100 UEContextReleaseRequest [Cause: RadioNetwork=user-inactivity]
  1772… 106.239561  172.16.0.6   172.16.0.4    NGAP        108 SACK (Ack=16, Arwnd=106496) , UEContextReleaseCommand [Cause: RadioNetwork=user-inactivity]
  1772… 106.240335  172.16.1.4   172.16.1.4    F1AP/N…     112 UEContextReleaseCommand, RRC Release MAC=0xf6c7cdcb
  1793… 106.745920  172.16.1.4   172.16.1.4    F1AP         88 UEContextReleaseComplete
  1793… 106.747091  172.16.0.4   172.16.0.6    NGAP         96 UEContextReleaseComplete
  2003… 111.872758  172.16.1.4   172.16.1.4    F1AP/N…     372 InitialULRRCMessageTransfer, RRC Setup Request
  2004… 111.876336  172.16.1.4   172.16.1.4    F1AP/N…     372 SACK (Ack=25, Arwnd=106496) , DLRRCMessageTransfer, RRC Setup
  2005… 111.917969  172.16.1.4   172.16.1.4    F1AP/N…     168 SACK (Ack=22, Arwnd=106496) , ULRRCMessageTransfer, RRC Setup Complete, Service request MAC=0x00000000
  2005… 111.918476  172.16.0.4   172.16.0.6    NGAP/N…     172 InitialUEMessage, Service request [RRCEstablishmentCause=mo-Data]
  2006… 111.921117  172.16.0.6   172.16.0.4    NGAP/N…     640 SACK (Ack=18, Arwnd=106496) , InitialContextSetupRequest
  2006… 111.924988  172.16.1.4   172.16.1.4    F1AP/N…     556 SACK (Ack=26, Arwnd=106496) , UEContextSetupRequest, Security Mode Command MAC=0xaf98d94f
  2006… 111.931343  172.16.1.4   172.16.1.4    F1AP        348 SACK (Ack=23, Arwnd=106496) , UEContextSetupResponse
  2007… 111.947940  172.16.1.4   172.16.1.4    F1AP/N…     108 ULRRCMessageTransfer, Security Mode Complete MAC=0xeb6899e6
  2007… 111.951215  172.16.1.4   172.16.1.4    F1AP/N…     360 DLRRCMessageTransfer, RRC Reconfiguration MAC=0x27f422f4
  2009… 111.987892  172.16.1.4   172.16.1.4    F1AP/N…     124 SACK (Ack=24, Arwnd=106496) , ULRRCMessageTransfer, RRC Reconfiguration Complete MAC=0x52c7cbd1
  2009… 111.988553  172.16.1.4   172.16.1.4    F1AP        108 SACK (Ack=29, Arwnd=106496) , UEContextModificationRequest
```

```
> Frame 5224: 312 bytes on wire (2496 bits), 312 bytes captured (2496 bits)
> Linux cooked capture v2
> Internet Protocol Version 4, Src: 172.16.1.4, Dst: 172.16.1.4
> Stream Control Transmission Protocol, Src Port: 38472 (38472), Dst Port: 38473 (38473)
> F1 Application Protocol (F1SetupRequest)
```

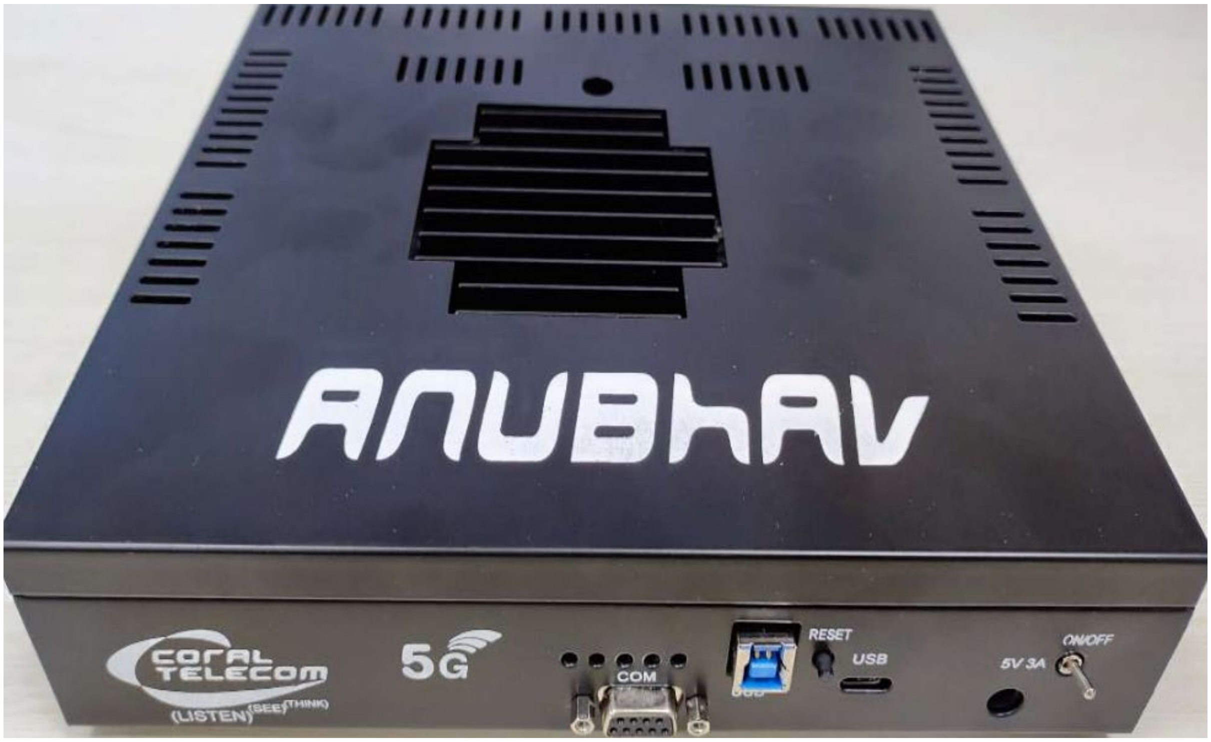
```
0000  08 00 00 00 00 00 00 01  03 04 00 06 00 00 00 00   ········ ········
0010  00 00 00 00 45 02 01 24  00 01 40 00 40 84 df 2a   ····E··$ ··@·@··*
0020  ac 10 01 04 ac 10 01 04  96 48 96 49 4a ce d3 38   ········ ·H·IJ··8
0030  00 00 00 00 00 03 01 01  04 78 f3 88 00 00 00 00   ········ ·x······
0040  00 00 00 3e 00 01 00 80  ec 00 00 05 00 4e 00 02   ···>···· ·····N··
0050  00 00 00 2a 00 02 00 01  00 2d 40 13 08 00 35 47   ···*···· ·-@···5G
0060  20 67 4e 42 2d 44 55 20  52 65 6c 20 31 2e 30 00    gNB-DU  Rel 1.0·
0070  2c 00 80 bc 00 00 00 2b  00 80 b5 4e 00 00 f1 10   ,······+ ···N····
0080  00 00 00 d0 50 00 0b 00  31 32 31 32 08 00 f1 10   ····P··· 1212····
0090  00 00 00 83 40 11 00 02  40 20 01 02 03 40 20 01   ····@··· @ ···@ ·
00a0  02 03 40 20 00 00 02 41  00 09 c4 b0 00 00 4d 00   ··@ ···A ······M·
```

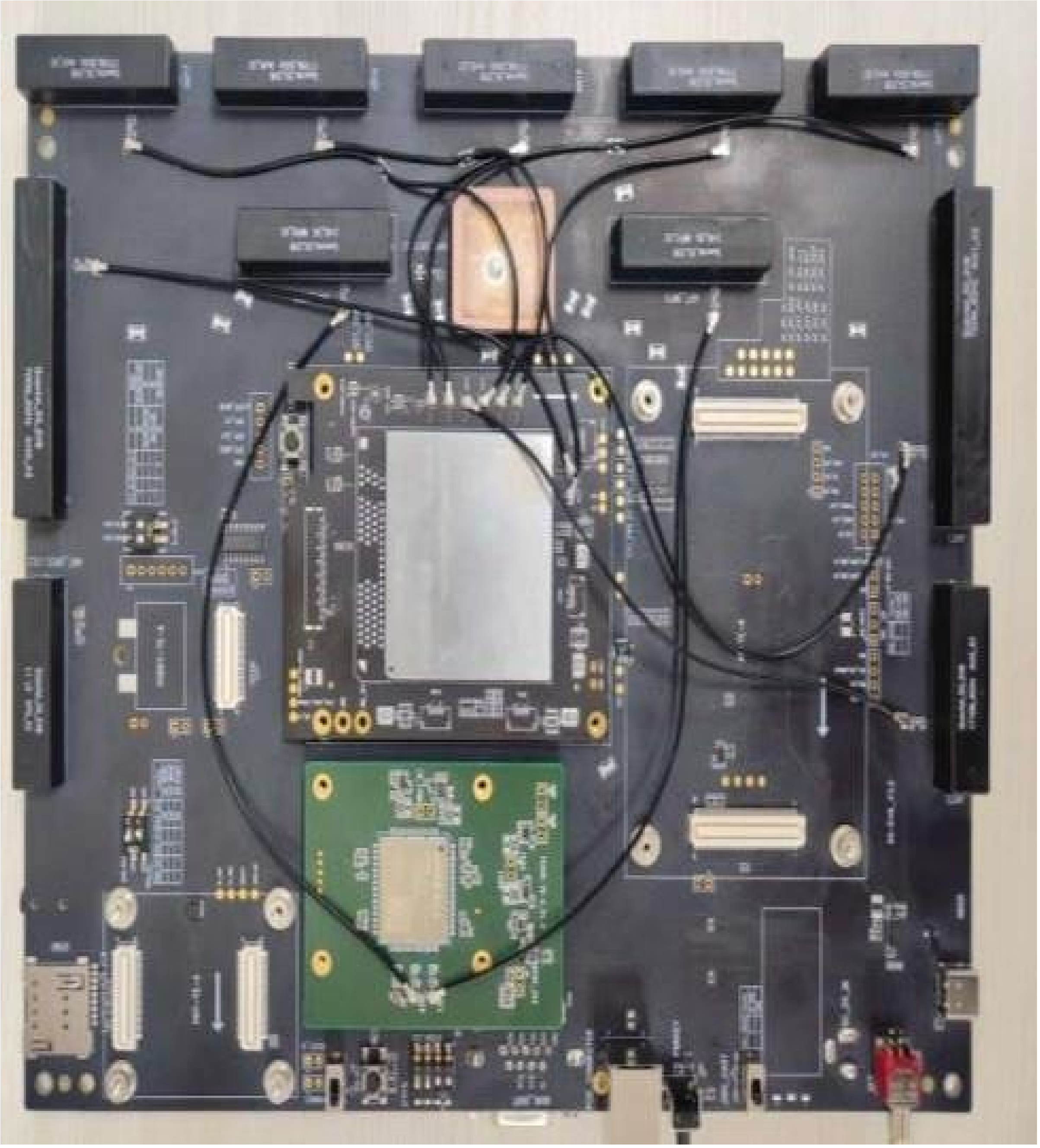
5G Lab Book

Part -4: 5G Use cases

Evaluation Board (EVB): ANUBHAV

The EVB is an auxiliary tool for engineers to develop and test various modules.





1. Introduction-

This document describes detailed information about the usage of the Coral Anubhav 5G EVB. The EVB is an auxiliary tool for engineers to develop and test these modules.

2. General Overview-

Coral Anubhav supplies 5G EVB for engineers to develop applications based on 5G and LTE-A EG512R-EA modules. This EVB can be used to test basic functionalities of various modules.

1. **Educational projects-**

   We can use the Coral Anubhav 5G EVB for educational projects related to telecommunications, wireless communication, and networking. They can explore topics such as signal processing, modulation schemes, multiple access techniques, and network protocols within the context of 5G.

2. **Prototyping and experiment-**

   The EVB allows to prototype and experiment with their own 5G-enabled devices and applications. They can develop IoT solutions, mobile applications, or other innovative projects that leverage 5G connectivity and advanced features.

3. **Research and development-**

   Pursuing research in areas like wireless networking, Internet of Things (IoT), smart cities, or augmented reality (AR) can utilize the EVB to conduct experiments and validate their hypotheses. They can explore how 5G technology can improve performance, reliability, and efficiency in various scenarios.

4. **MIMO-**

   Multiple-Input Multiple-Output (MIMO) is a wireless technology that uses multiple transmitters and receivers to transfer more data at the same time.

Key features of Coral Anubhav

Features	Implementation
Power Supply	DC power supply: 4.5–5.5 V Typical: +5 V/ 3 A
Module TE-A Interface	5G RG50xQ Series and LTE-A EG512R-EA modules supported
PHY TE-A Interface	PHY AR8035 supported
Wi-Fi TE-A Interface	Reserved
AP TE-A Interface	Reserved
SD Card Interface	SD card connector
(U)SIM Card Interface	Dual (U)SIM card supported: 1.8 V and 2.95 V
Audio Interfaces	• 1 digital audio codec board interface: Supporting ALC5616-TE-A and TLV320AIC3104-TE-A codec boards • 2 analog audio interfaces: Used for loudspeaker and earphone
UART Interface	COM1 (J2002): • Main UART • For data communication • Default baud rate: 115200 bps COM2 (J2003): • Debug UART • For debugging • Default baud rate: 115200 bps
USB Interface	• USB Type C interface • USB 3.0 and USB 2.0 supported
PCIe to USB Interface	Reserved
Signal Indication	5 LEDs are available for signal indication
Button and Switches	• PWRKEY (S0202) • RESET (S0201) • USB_BOOT (S0203) • PCIe Configuration Switch (S1501) • Power Switch (S0301) • SDIO Configuration Switch (S2501) • RGMII Configuration Switch (S1801/S1802) • Codec Configuration Switch (S2801
Physical Characteristics	Size: 235 mm ×190 mm
Antenna	12 antenna interfaces

1. Interface Application

This section describes the hardware interfaces of Coral Anubhav 5G EVB, as listed below:

- Power supply
- Module TE-A interface
- PHY TE-A interface
- USB interface
- Audio interfaces –
  - Digital Audio Codec Board Connector
  - -Analog Audio Interfaces
    - -Loudspeaker Interface
    - -Earphone Interface
- (U)SIM interfaces
- UART interfaces
- SD card interface
- PCIe to USB interface
- Switches and buttons
- Status indicators
- Wi-Fi interface
- Antenna Interfaces

1. Power Supply

Coral Anubhav EVB can be powered by an external power adapter through the power jack (J0303).

EVB Power Supply Interface

2. **USB Interface (J1101)**
   A USB Type C connector, which complies with USB 3.0/3.1 and USB 2.0 standard, is provided. This USB interface is used for AT command communication, data transmission and firmware upgrade.

USB Interface Connection

3. **Audio Interface (J0802/J0901/J0801)**
   Coral Anubhav provides two analog audio interfaces J0901 and J0801.

Analog Audio Interfaces (J0801/J0901)

Audio interface J0901 is designed for earphone. A reference circuit design is shown by the following figure

Earphone interface

4. **(U)SIM Card Interfaces (J1401/J1402)**
   Coral Anubhav 5G EVB has two 8-pin push-push type (U) SIM card (1.8/2.95 V) connectors J1401 and J1402. The figure below illustrate the pin assignment and definition of (U) SIM card connector J1401. J1402 is similar to J1401.

Pin Assignment of (U) SIM Card Connector J1401

5. **SD Card Interface (J1301)**
   Coral Anubhav 5G EVB provides an SDIO interface, which can be used for connecting SD card. The following figure shows the simplified interface schematic for J1301. If SD card function is intended to be used, please switch the SDIO Switch to low level illustrated in the figure below, a standard SD card can be inserted into J1301. Which supports micro-SD card of maximal 32 GB. With the SD card interface, customers can easily enhance the memory capacity of modules.

SD Card Connector J1301

6. UART Interfaces (J2002/J2003)

- Coral Anubhav supports two UART interfaces: main UART J2002 and debug UART J2003, supporting baud rate of 115200 bps by default.
- The main UART interface is used for communication between the module and the host application.
- The debug UART interface is used for Linux console and log output.

Main UART Interface (J2002)

7. PCIe to USB Interface (J1601)

This EVB reserves a PCIe 3.0 signal over USB interface for developers’ testing, and this function is not enabled by default.

PCIe to USB Interface

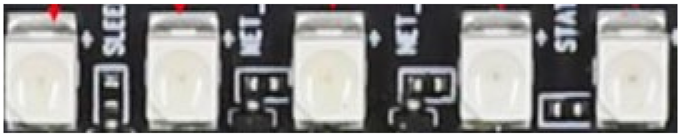
8. Switches and Buttons

Coral Anubhav includes one switch S0301 and one button, as shown in the following figures.

Switch and buttons

9. Status Indicators (D0201/D0202/D0203/D0204/D0205)

There are five status indication LEDs on the EVB. The following figure indicates the positions of these LED indicators.



Status Indicator

10. Antenna Interfaces

Coral Anubhav provides twelve antenna interfaces. The following displays the assembly of these antenna interfaces:

Antenna No.	Frequency Range (Hz)
ANT0, ANT1	700 MHz–6 GHz
ANT2–ANT7	1.7 GHz–6 GHz
ANT GNSS L1	GNSS L1
ANT GNSS L1 L5	GNSS L1 L5
ANT Wi-Fi 0 & 1	2.4–5 GHz

Cellular Antenna

GPS & Wi-Fi Antenna

Coral Anubhav Operation Procedures

This introduces how to use the 5G EVB for testing and evaluation of Quectel modules.

Turn on the Module

1. Connect the module TE-A to the EVB via connectors J0101 and J0102.
2. Insert a (U) SIM card into the USIM1 card connector on EVB.
3. Use RF cable to connect the module TE-A to the EVB, and connect antennas to the EVB.
4. Connect the EVB to a 5 V/ 3 A power, then switch S0301 to ON. Then D0201 (ON/OFF indicator of the module’s power supply) will light up.
5. Press the S0202 (PWRKEY) for at least 500ms, then the module will be powered on and D0202 (operation indicator of the module) will light up.

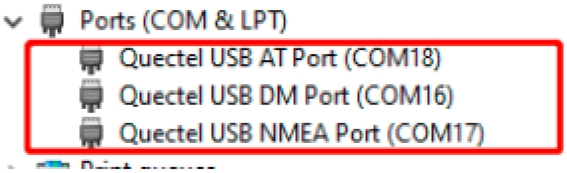
Turn Off the Module

There are two methods to turn off the module.

- Turn off the module with AT+QPOWD. This is a safe method. The module will log off from the network and save data before shutdown.
- Turn off the module with PWRKEY button (S0202). Long press PWRKEY for at least 800 ms and the module will be turned off.

Communication via USB

1. Power on the module according to the procedure in upper section
2. Connect the EVB and a PC with USB cable through USB Type-C interface, and then run the driver disk on the PC to install the USB driver. For details about USB driver installation, please refer to document provided. The USB port numbers can be viewed in Device Manager of the PC when the USB driver is installed, as shown below:



Ports (COM & LPT)
Quectel USB AT Port (COM18)
Quectel USB DM Port (COM16)
Quectel USB NMEA Port (COM17)

USB Ports

Firmware Upgrade

Firmware of the module is upgraded via USB port by default, and there are two methods for the upgrade: forced download and normal download. See the following procedures to upgrade firmware through the EVB.

Forced Download

- Install the firmware upgrade tool Flash on PC.
- Connect the EVB and the PC through USB Type-C cable.
- Insert the DC power adapter.
- Press the USB_BOOT button (S0203) and then turn on the module.

Normal Download

- Turn on the module according to the procedures.
- Wait for the USB port to be found in “Device Manager” of the PC

Steps To Start With Evaluation Board

Step 1: -Insert the Private 5G SIM in the Evaluation Board
Step 2: - Power on the Board using 5V Power Adapter
Step 3: - Press the Reset Button for 3-5 Seconds
Step 4: - Wait for 15 seconds for the Board to Boot Up
Step 5: - Install the appropriate drivers on the computer.

Recommended: Use Driver Booster: Driver Booster .

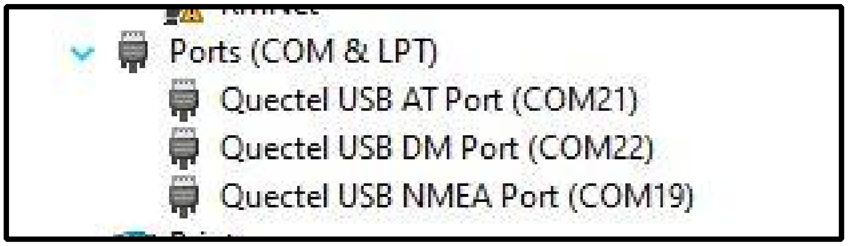
- Install the RG520F Drivers (Must Needed)
- Install the PORT Driver (Must Needed)

You are now ready to take the access of the evaluation board

Basic Configuration of Evaluation Board

Step1: Press Windows + X on your Windows Computer Home screen and navigate to Device Manager

Step2: When you will open the Device Manager you see a list of Ports(COM & LPT)



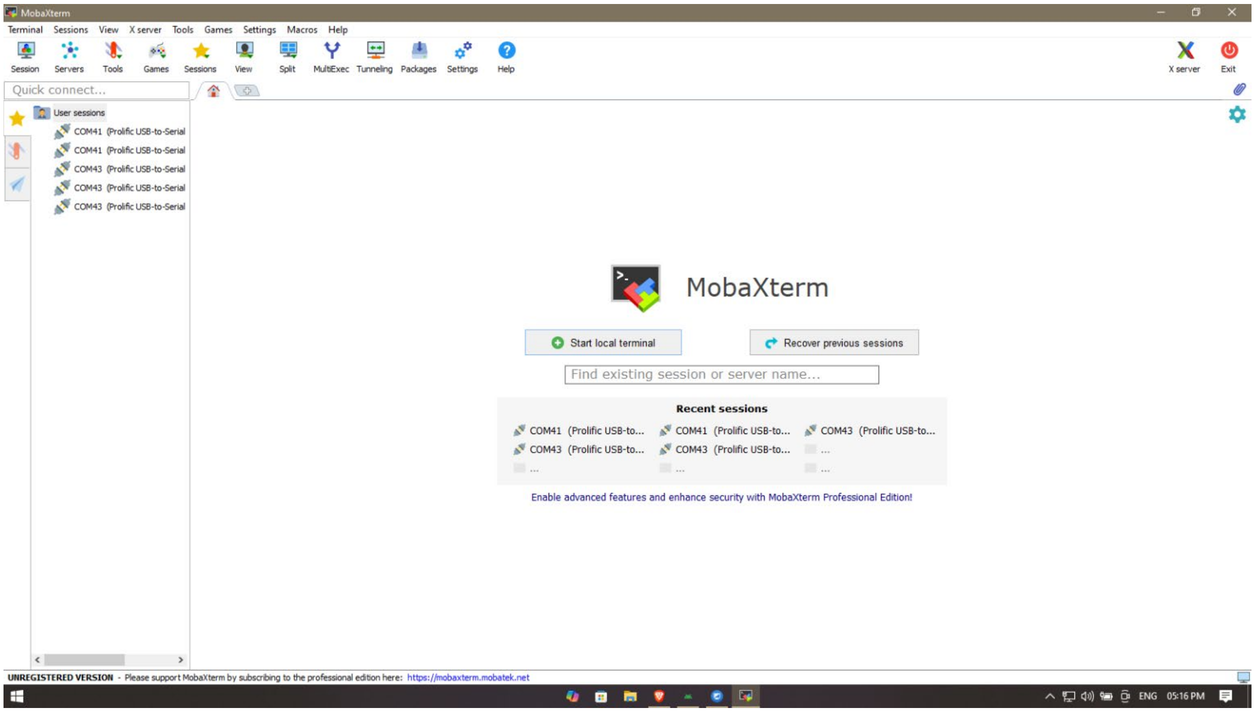
Ports (COM & LPT)
Quectel USB AT Port (COM21)
Quectel USB DM Port (COM22)
Quectel USB NMEA Port (COM19)

Step 3: - After that install MobaXterm from the internet

Step 4: - MobaXterm is an advanced terminal emulator and remote desktop application designed for Windows. It provides a powerful set of tools for developers, system administrators, and IT professionals who work with remote systems

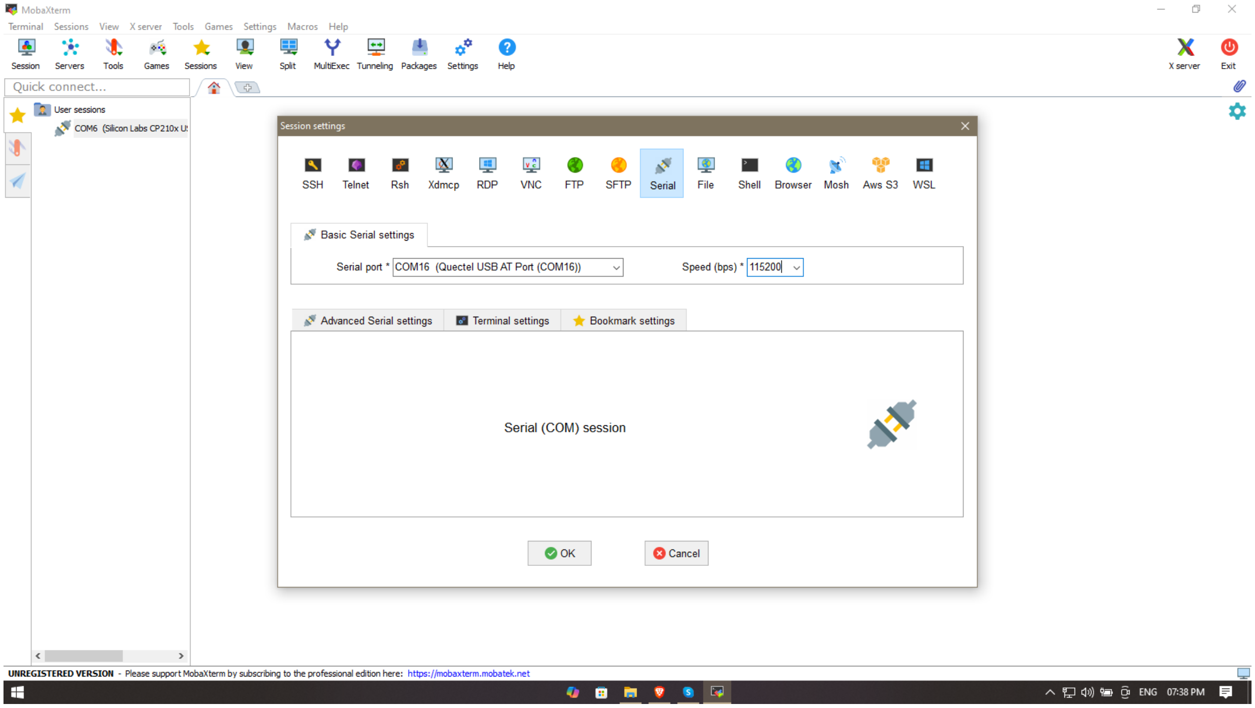
Steps To Access Coral Anubhav with MobaXterm:

- When You Open MobaXterm, you will be welcomed with the screen.
- Select the option of Session on the Top-Right Corner of the Screen.
- Once you click on that you will see another window.
- Once you get to this screen, go for serial connection which have the logo of plug connector.
- In This Screen you Select the AT Port and enter the BAUD Rate as 115200.
- After all this you will be able to access the Coral Anubhav.
- Type ATI to see the Basic Details:
-


MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
User sessions
COM41 (Prolific USB-to-Serial
COM41 (Prolific USB-to-Serial
COM43 (Prolific USB-to-Serial
COM43 (Prolific USB-to-Serial
COM43 (Prolific USB-to-Serial
MobaXterm
Start local terminal
Recover previous sessions
Find existing session or server name...
Recent sessions
COM41 (Prolific USB-to...
COM41 (Prolific USB-to...
COM43 (Prolific USB-to...
COM43 (Prolific USB-to...
COM43 (Prolific USB-to...
...
...
...
...
ENG 05:16 PM

Enable advanced features and enhance security with MobaXterm Professional Edition!

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net


MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
User sessions
COM6 (Silicon Labs CP210x U:
Session settings
SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh Aws S3 WSL
Advanced Serial settings
Terminal settings
Bookmark settings
Serial (COM) session
OK
Cancel
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
ENG 07:38 PM

Basic Serial settings

Serial port * COM16 (Quectel USB AT Port (COM16))

Speed (bps) * 115200

Introduction To Attention Commands (AT Commands)

AT commands (Attention commands) are used to communicate with modems or modules via serial communication. They help configure settings, retrieve information, or control device functionalities.

Types of AT Commands

Test Command (=?) - The command returns the list of parameters and value ranges set by the corresponding Write Command or internal processes.

Syntax: AT+COMMAND=?

Purpose: Lists all possible values a command supports.

Example Response: +CONFIG: (0,1,2,3), indicating valid options.

Read Command (?) - The command returns the currently set value of the parameter or parameters.

Syntax: AT+COMMAND?

Purpose: Queries the current value of a setting.

Example Response: +CONFIG: 2, meaning the current setting is 2.

Write Command (=<parameters>) - The command sets the user-definable parameter values.

Syntax: AT+COMMAND=<value>

Purpose: Sets a new value for the configuration.

Example Response: OK, confirming the update.

AT Command	Functionality
ATI	**Returns model number and firmware version.**
AT+CIMI	**Returns IMSI number.**
AT+COPS=?	**Displays list of available networks; check if network "00101" is available.**
AT+CFUN=0	**Switches UE to minimum functionality (returns OK).**
AT+CFUN=1	**Switches UE to full functionality (returns OK).**
AT+CGDCONT?	**Displays list of APNs; check APN configuration as per network slice.**
AT+CGDCONT=1,"IP","APN-Name"	**Set APN to "APN-Name".**

Configuration Of Coral Anubhav (5G Evaluation Board) For 5G Registration

Now, look at how to connect your 5G Evaluation board with the radio and establish 5G Registration.

Entire registration part is divided into 4 Steps

Step 1: Set network mode preference to NR5G

Set Network Preferences: AT+QNWPREFCFG="mode_pref",NR5G

Step 2: Set APN to 'APNname' for data connection.

Set APN: AT+CGDCONT=1,"IP","APNname"

Step 3: Set the operator selection on Automatic mode

Set operator selection on automatic mode: AT+COPS=0

Step 4: Reload The 5G Evaluation Board (Set UE Functionality)

```
AT +CFUN=0
AT+CFUN=1
```

Raspberry Pi Use Case with Coral Anubhav

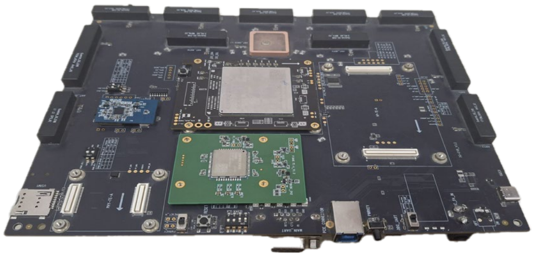
Introduction

This use case explores the practical applications of automating the configuration of the 5G evaluation(Coral Anubhav) using Bash scripts on a Raspberry Pi. By leveraging predefined commands, users can efficiently manage network registration, modem setup, and configuration changes without manual intervention. This automation enhances reliability, speeds up deployment, and reduces human errors in 5G connectivity setups, making it ideal for IoT applications, industrial automation, and research projects. The guide provides a step-by-step approach to executing AT commands via Bash, ensuring seamless interaction between the Raspberry Pi and the 5G modem.

Prerequisites

- Raspberry Pi (any model with USB support)
- 5G Evaluation Board (Coral Anubhav)
- SIM Card
- Minicom or another serial communication tool

USB-to-Serial driver installed (if necessary)



Setup Instructions

Run as Root:

```
sudo -i
```

Check Device Path:

```
ls /dev/ttyUSB*
```

Install Required Packages:

```
sudo apt update && sudo apt install -y socat;
```

Grant USB Permissions:

```
sudo chmod 777 /dev/ttyUSB*
```

Bash Script for Modem Configuration

Create a script `modem_config.sh` to execute AT commands using `socat`.

```
#!/bin/bash

# Check if minicom is installed

if ! command -v socat &> /dev/null; then

    echo "socat is not installed. Installing..."

    sudo apt update

    sudo apt install -y socat

else

    echo "script is started."

fi

# The command run and output show in file

echo ATI | socat - /dev/ttyUSB2,crnl > /tmp/hello;

echo AT+cimi | socat - /dev/ttyUSB2,crnl > /tmp/hello;

echo AT+QNWCFG=? | socat - /dev/ttyUSB2,crnl >/tmp/hello;

#when you want to append the output

#echo AT+QNWCFG=? | socat - /dev/ttyUSB2,crnl >>/tmp/output.txt;
```

```
# for print on file and terminal

#echo ATI | socat - /dev/ttyUSB2,crnl | tee /tmp/hello;

#    read command thru file and output both to terminal and file

#cat commands.txt | socat - /dev/ttyUSB2,crnl | tee /tmp/hello
```

Running the Script

Make the script executable and run it:

```
chmod +x modem_config.sh

sudo ./modem_config.sh
```

Creating a Systemd Service for Automation

To automate the execution of the script, create a systemd service.

```
[Unit]
Description=5G Modem Configuration Service
After=network.target

[Service]
ExecStart=/bin/bash /path/to/modem_config.sh
Restart=always
User=root

[Install]
WantedBy=multi-user.target
```

Save the file as `/etc/systemd/system/modem_config.service` and run the following commands to enable it:

```
sudo systemctl daemon-reload
```

Reloads systemd to recognize new or modified service files.

```
sudo systemctl enable modem_config.service
```

Enables the service to start automatically on boot.

```
sudo systemctl start modem_config.service
```

Starts the service immediately without rebooting.

Trouble Shooting of Coral Anubhav

Basic Checks

- Power & Hardware Connections
- Ensure the module is properly powered
- Check if the antennas are connected (for optimal signal reception)
- Verify the SIM card is inserted correctly
- Driver & Firmware Verification
- Check if the necessary drivers are installed (Linux/Windows)
- Ensure firmware is up-to-date
- Default Baud Rate as 115200. Evaluation Board will communicate with devices on this BAUD Rate

AT Command Interface Check

Use AT commands to verify basic functionality

- AT → Check if the module responds
- ATI → Get manufacturer info
- AT+CGMR → Get firmware version

Debugging Logs

- Insert a USB Type-C connection with Debug UART and then switch to the Silicon Labs CP210x USB-to-UART Bridge. Set the BAUD Rate as 115200.This will provide access to hardware-level logs and detailed diagnostic information for all modules and features.

Device and Module Info

➔ Get Firmware and Device Information:

AT+CGMR - Display firmware version.

➔ Check 5G MIMO Status:

AT+QNWCFG="nr5g_mimo" -Check if 5G MIMO is enabled.

➔ Enable 5G MIMO:

AT+QNWCFG="nr5g_mimo",1-Enable 5G MIMO.

ADB (Android Debugging Bridge) Access

- To check about the Coral Anubhav services, we can access it by taking adb access to the Coral Anubhav Board.
- Take access by adb shell after connecting it via USB 3.0 provided
- Enter the command - systemctl status *.service

Using Evaluation Board for Use Case Projects in Smart Homes, Smart City etc

Introduction

- Raspberry Pi as a low-cost, versatile device for IoT applications.
- Used in **Smart Homes, Smart Cities, and Smart Agriculture** to collect and transmit sensor data
- You can use smoke sensor, gas leak sensor & fire detection sensor etc.

How It Works

- Raspberry Pi connects to various sensors (Smoke Sensor, Gas Leak Sensor & Fire Detection Sensor etc)
- Data is sent to a cloud/server via Wi-Fi or a 5G Evaluation Board (Coral Anubhav)
- Enables automation and real-time monitoring for smart applications.

Setup & Connectivity

- Raspberry Pi is configured to connect to WiFi or a 5G evaluation board.
- Sensors are connected to the Raspberry Pi.

Data Processing & Transmission

- A **Python script** can be used to collect real-time sensor data.
- Data is transmitted to a **server/NMS**
- The server processes and stores the data for analysis.

Applications & Benefits

- Smart Homes: Automates appliances, security, and climate control.
- Smart Cities: Air quality monitoring, traffic optimization, and energy management.
- Smart Agriculture: Monitors soil moisture, controls irrigation, and optimizes farming.

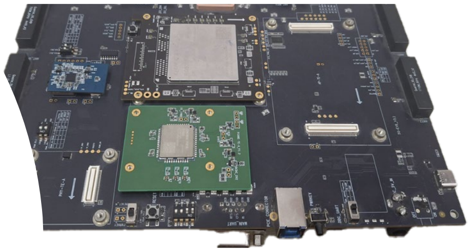
Automating 5G Evaluation Board Configuration with Bash Scripts

Overview

- Automates 5G Evaluation Board setup on Raspberry Pi using Bash scripts.
- Eliminates manual configuration, ensuring faster deployment & reduced errors.
- Ideal for IoT applications, industrial automation, and research projects.

Prerequisites

- Raspberry Pi with USB support.
- 5G Evaluation Board & SIM card.
- Minicom / Socat for serial communication.
- USB-to-Serial driver (if needed).



Setup Steps

1. Run as Root
   ```
   sudo -i
   ```
2. Check Device Path
   ```
   ls /dev/ttyUSB*
   ```
3. Install Required Packages
   ```
   sudo apt update && sudo apt install -y socat
   ```
4. Grant USB Permissions
   ```
   sudo chmod 777 /dev/ttyUSB*
   ```

Key Features of Automation

Hands-free Setup – No need for manual command execution.
Faster Deployment – Reduces setup time for 5G connectivity.
Error Reduction – Minimizes human mistakes in modem configuration.
Auto-Restart on Failure – Systemd ensures continuous operation.

How It Works

1. Raspberry Pi runs predefined Bash commands to configure the modem.
2. The script automates network registration & modem setup.
3. A systemd service ensures the script runs automatically on boot.
4. The modem continuously operates without manual intervention.

Conclusion

- Automating 5G modem setup improves efficiency & reliability.
- Reduces downtime & ensures seamless IoT & industrial automation.
- Easy to implement using Bash scripting & systemd services.

IoT Sensor

Introduction to IoT Sensor Software (Coral Gyan)

1. Overview

The IoT Sensor Software (Coral Gyan) is a cutting-edge device designed to bring the power of the Internet of Things into your hands. This compact and versatile gateway serves as a hub for connecting various smart devices and sensors, allowing you to seamlessly integrate them into a unified and intelligent system.

2. Purpose

In an era where connectivity is key, the IoT Sensor Software (Coral Gyan) is crafted to simplify the way you interact with your environment. It empowers users to monitor, control, andgather data from a myriad of IoT-enabled devices, fostering a matter and more efficient living or working space.

3. Key Features

3.1 Connectivity

The Device can be connected to IoT Gateway or 5G CPE via Ethernet.

3.2 Seamless Integration

With its intuitive interface and compatibility with popular IoT platforms, the device facilitates the integration of devices from different manufacturers. This interoperability allows users to create a cohesive and interconnected network of smart devices.

3.3 User-Friendly Interface

Designed with simplicity in mind, the user interface of the device is user-friendlyand accessible. Whether you are a tech enthusiast or a beginner, you can effortlessly configure and manage your IoT devices with ease.

4. Applications

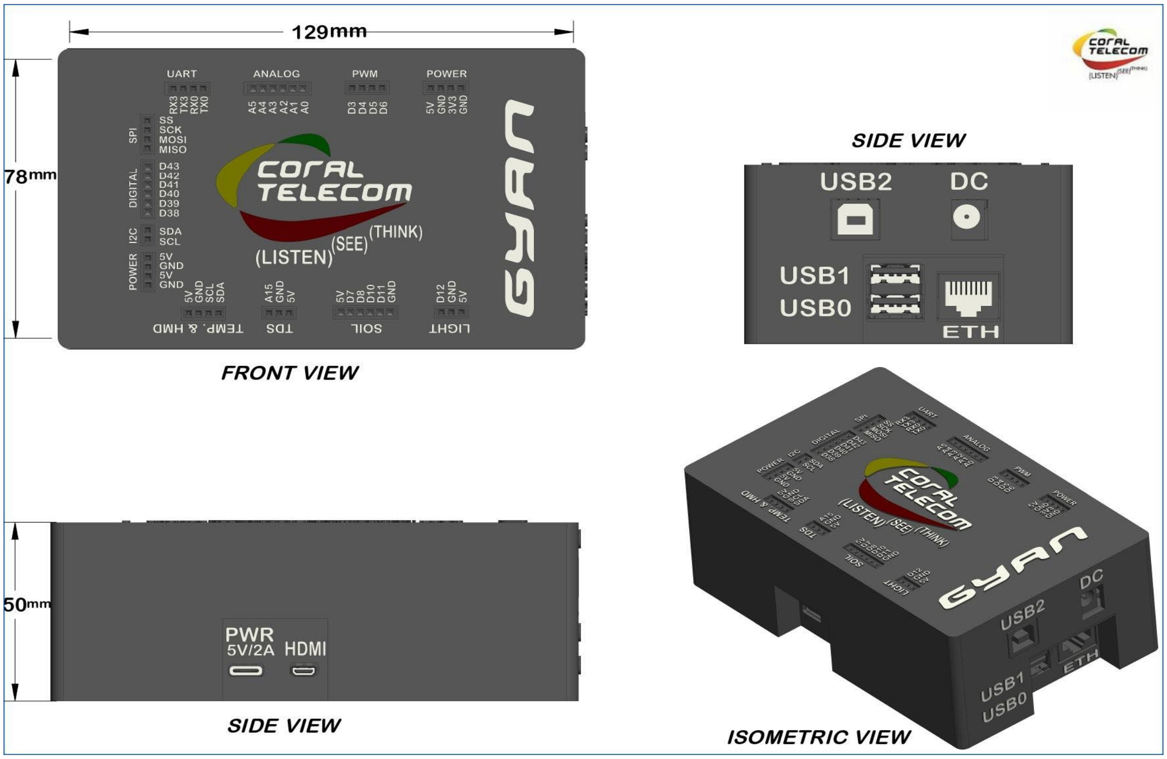
The IoT Sensor Software (Coral Gyan) is a versatile microcontroller board that can be used in various applications. Its extended capabilities, including larger I/O pins, make it suitable for complex projects. Here are some common applications and uses of:

- Smart Home Automation: Control lights, thermostats, cameras, and other devices remotely.
- Industrial IoT: Monitor and manage equipment, collect data, and streamline operations.
- Healthcare: Enable remote health monitoring and data collection for patients.
- Environmental Monitoring: Track and analyze environmental parameters such as temperature, humidity, and air quality.
- Prototyping and Development: it is commonly used for prototyping and developing electronic projects. Its ease of use and vast community support make it an excellent choice for hobbyists, students, and professionals.
- Robotics: It is widely used in robotics projects due to its extensive I/O capabilities. It can control multiple motors, sensors, and communication modules, making it suitable for building advanced robotic systems. With its numerous analog and digital pins, the IoT Sensor Software (Coral Gyan) is well-suited for building data acquisition systems. It can collect and process data from various sensors, such as temperature sensors, humidity sensors, and accelerometers.

Getting Started

Getting started with the IoT Sensor Software (Coral Gyan) is a simple process. Follow the included setup guide to connect the device to your network and begin adding your IoT devices to the system. Whether you're looking to enhance your living space, optimize business operations, or delve into the world of IoT development.

Dimensions of IoT Sensor Software (Coral Gyan):
"All details about dimensions are shown in the figure below."



The power pins are as follows:

- 5V: The regulated power supply is used to power the microcontroller and other components on the board. The Coral Gyan Box can be powered by using 5V USB adapter.
- 3V3: A 3.3-volt supply generated by the on-board regulator. The maximum current draw is 50 mA.
- GND: Ground pins.
- SPI: (MISO), (MOSI), (SCK), (SS). These pins support SPI communication.
- Serial: 0 (RX) and 1 (TX); Serial 3: (RX) and (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- PWM: 3 to 6. Provide 8-bit PWM.
- (SDA) and (SCL). Support I2C (TWI) communication.
- Connection to power on a 5.1V/2.5A DC output Type-B charger is provided to power it up.
- Serial Data Connection: For serial data communication, a serial cable is provided, which can be connected on board USB.

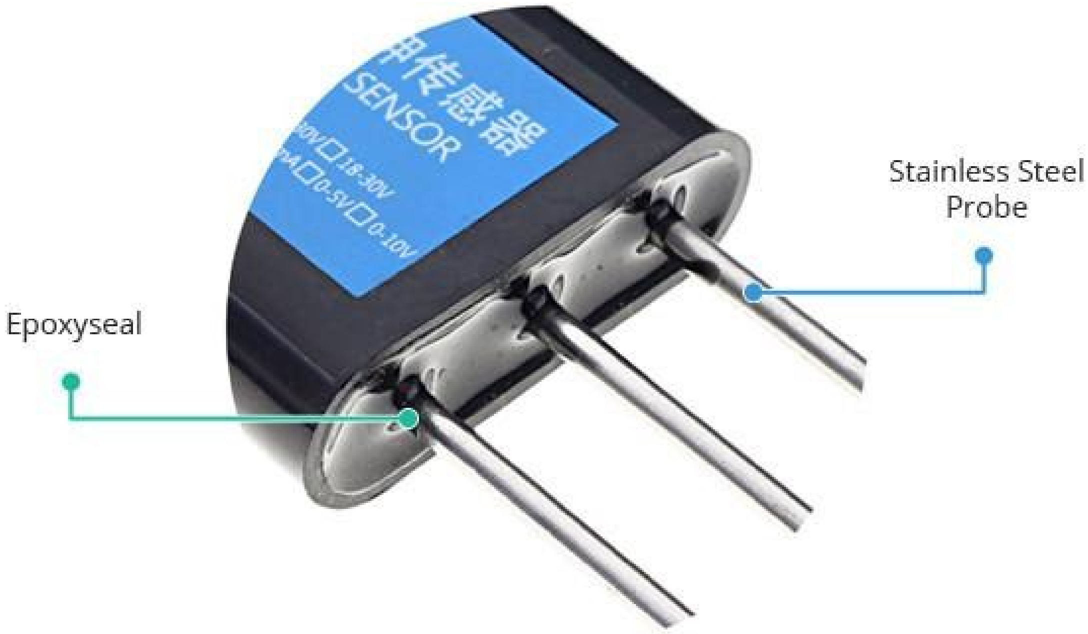
Soil NPK sensor

The soil NPK sensor is suitable for detecting the content of nitrogen, phosphorus, and potassium in the soil, and judging the fertility of the soil. thereby facilitating the systematic evaluation of the soil condition. It can be buried in the soil for a long-time, resistant to long-term electrolysis, corrosion resistance, vacuum potting, and completely waterproof. Soil NPK sensors are widely used in soil nitrogen, phosphorus and potassium detection, precision agriculture, forestry, soil research, geological prospecting, plant cultivation and other fields.

The soil NPK sensor can detect the levels of nitrogen, phosphorus, and potassium in the soil (not in water). It helps determine soil fertility, allowing for a more systematic assessment of soil condition.

NPK soil sensor

The sensor operates on 5-30V and consumes very little power. According to the datasheet, it is capable of measuring nitrogen, phosphorus, and potassium with aresolution of up to 1 mg/kg (mg/l). The sensor includes a stainless-steel probe that is rust-proof, electrolytic-resistant, and salt-alkali resistant. It can therefore be used with any type of soil,including alkaline soil, acid soil, substrate soil, seedling bed soil, and coconut bran soil.



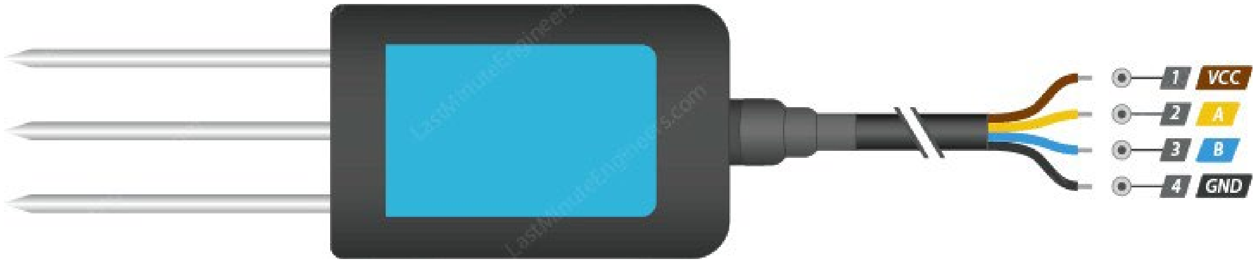
The probe is sealed to the body with high-density epoxy resin to prevent moisture from entering the body. The best part is that the sensor has an IP68 rating, which means it is protectedagainst dust and moisture, allowing it to operate normally for a very long time.

To be used effectively over long distances, the sensor features the RS485 communication interface and supports the standard Modbus-RTU communication protocol.

Technical Specifications:

Soil NPK Sensor Pinout

The sensor comes with a 2m cable with tinned copper wires. The pinout is shown in the figure below.



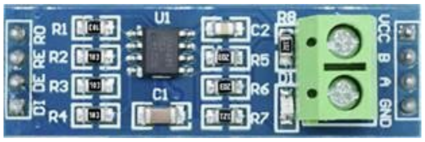
NPK Sensor Pinout

VCC is the VCC pin. Connects to 5V.
A is a differential signal that is connected to the A pin of the MAX485 Modbus Module.

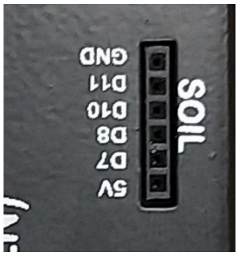
B is another differential signal that is connected to the B pin of the MAX485 Modbus Module.

GND is the Ground pin.

MAX485 Modbus Module



Interface of Soil NPK Sensor with Coral Gyan



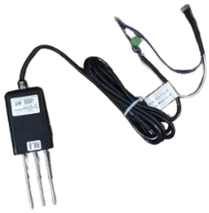


Figure 1: NPK Sensor Slot Figure 2: NPK Sensor Figure 3: Coral Gyan

As shown in Figure 2 connect the NPK Sensor (Soil Sensor) on the respective Slot shown in Figure 1 on the Coral Gyan Box shown in Figure 3, While connecting the NPK sensor to the Coral Gyan box on the desired slot, check the sensor pins for white marking. That marking will indicate from which side the pin will be connected to the sensor slot. The white marking on the pin indicates that the pin will be connected to the 5V mark (as shown in Figure 1) on the Coral Gyan Box.

Temperature & Humidity Sensor (AHT25)

AHT25 is a quick-response, calibrated, reliable temperature and humidity sensor which permits the measurement of environmental parameters such as humidity and temperature.

pin description left to right side

- VDD - connected to 5.5V
- SDA -Serial Data Bidirectional port
- GND - connected to ground
- SCL - Serial clock Bidirectional port (pin no 4 connected to SCL pin)

Interface of Temperature & Humidity (AHT25) Sensor with Coral Gyan

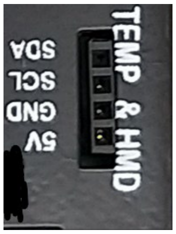


Figure: Temp&HMD Slot

Figure: Temperature Humidity

Figure: Coral Gyan

As shown in Figure connect the Temperature & Humidity Sensor (AHT25) on the respective Slot shown in Figure on the Coral Gyan Box shown in Figure, while connecting the temperature & humidity sensor to the Coral Gyan box on the desired slot, check the sensor pins for white marking. That marking will indicate from which side the pin will be connected to the sensor slot. The white marking on the pin indicates that the pin will be connected to the 5V mark (as shown in Figure) on the Coral Gyan Box.

TDS sensor

Specification

- Working Current: 3 ~ 6mA
- TDS Measurement Range: 0 ~ 1000ppm
- TDS Measurement Accuracy: ± 10% F.S. (25 ℃)
- Module Size: 42 * 32mm
- Module Interface: PH2.0-3P
- Electrode Interface: XH2.54-2P

TDS probe

- Number of Needle: 2
- Total Length: 83cm
- Connection Interface: XH2.54-2P
- Colour: Black
- Other: Waterproof Probe

Total Dissolved Solids (TDS) of water are directly related to the conductivity of dissolved ionized solids in the water. Ions from the dissolved solids create the ability for water toconduct an electrical current, which is measured by the TDS sensor in ppm.

The Interface of TDS Sensor with Coral Gyan:

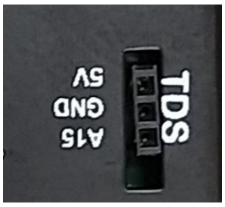


Figure1: TDS Slot

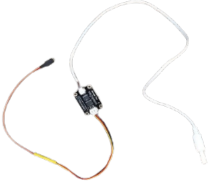


Figure2: TDS Sensor

Figure 3: Coral Gyan

As shown in Figure 2 connect the TDS Sensor (TDS Sensor) on the respective Slot shown in Figure 1 on the Coral Gyan Box shown in Figure 3, While connecting the TDS sensor to the Coral Gyan box on the desired slot, check the sensor pins for white marking. That marking will indicate from which side the pin will be connected to the sensor slot. The white marking on the pin indicates that the pin will be connected to the 5V mark (as shown in Figure 1) on the Coral Gyan Box.

LDR Sensor

LDR means Light detection or dependent resistor sensor which is capable of measuring and detecting light intensity. It is also known as a photoresistor sensor. It helps to detect light. This sensor module comes with 3 terminals. Where the “DO” pin is a digital output pin.

Pin No	Pin Name	Description
1	VCC	+5 v power supply Input Pin
2	GND	Ground (-) power supply Input Pin
3	DO	Digital Output Pin

Working Principle:

The Light Dependent Resistor (LDR) works on the principle of “Photo Conductivity”. The LDR resistance is changed according to the light intensity that falls on the LDR. When light intensity increases on the LDR surface, then the LDR resistance will decrease and the element conductivity will increase. When light intensity decreases on the LDR surface, then the LDR resistance will increase and the element conductivity will decrease.

The Interface of LDR Sensor with Coral Gyan:

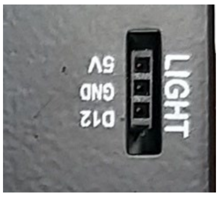


Figure1: LDR Slot

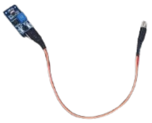


Figure2: LDR Sensor

Figure 3: Coral Gyan

As shown in Figure 2 connect the LDR Sensor (LDR Sensor) on the respective Slot shown in Figure 1 on the Coral Gyan Box shown in Figure 3, While connecting the TD sensor to the Coral Gyan box on the desired slot, check the sensor pins for white marking. That marking will indicate from which side the pin will be connected to the sensor slot. The white marking on the pin indicates that the pin will be connected to the 5V mark (as shown in Figure 1) on the Coral Gyan Box

The connection of sensors to IoT Sensor Software (Coral Gyan) is written on the box.

Introduction - These manual talks about showing your sensor data to your NMS.

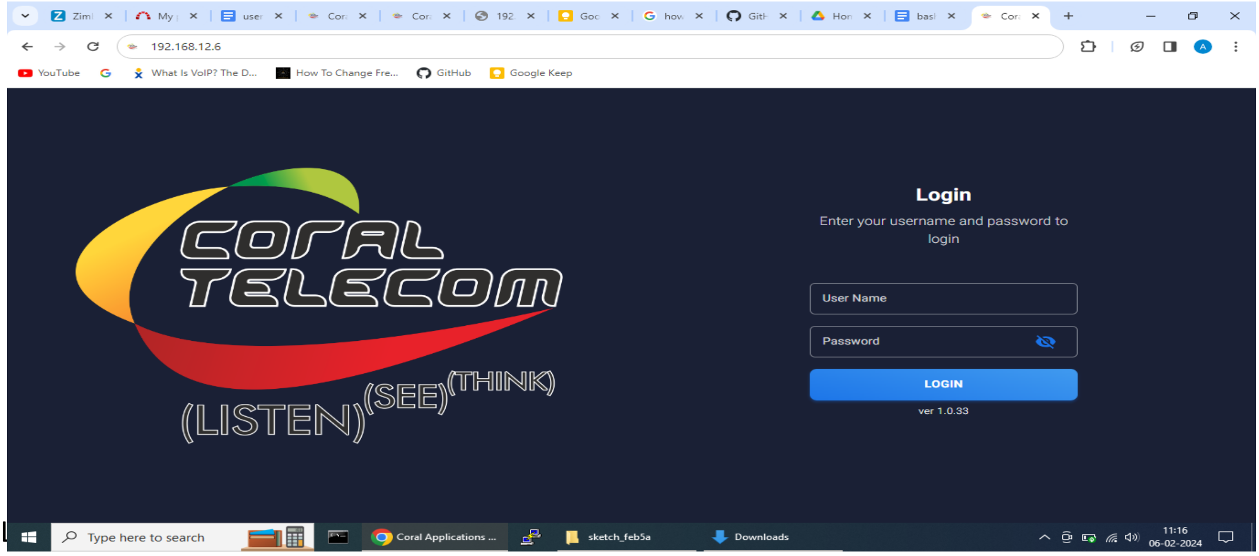
- STEP 1- FIRST CONNECT WITH NETWORK.
- STEP 2- CREATE A USER
- STEP 3 - LOGIN WITH THE CREATED USER
- STEP 4 - CREATE A DASHBOARD FOR YOUR SENSOR DATA
- STEP 5 - ADD WIDGET FOR SENSOR
- STEP 6 - ADD WIDGET GRAPH TO YOUR DEFAULT PAGE

STEP 1- CONNECT YOUR PC/LAPTOP TO THE LAB NETWORK USING 5G CPE OR ETHERNET CABLE.

STEP 2 - CREATE A USER

For creating a user, you (student/learner) want to log NMS from your browser NMS IP or domain “192.168.12.6”

You may see this type of pages:



Login

Enter your username and password to login

User Name

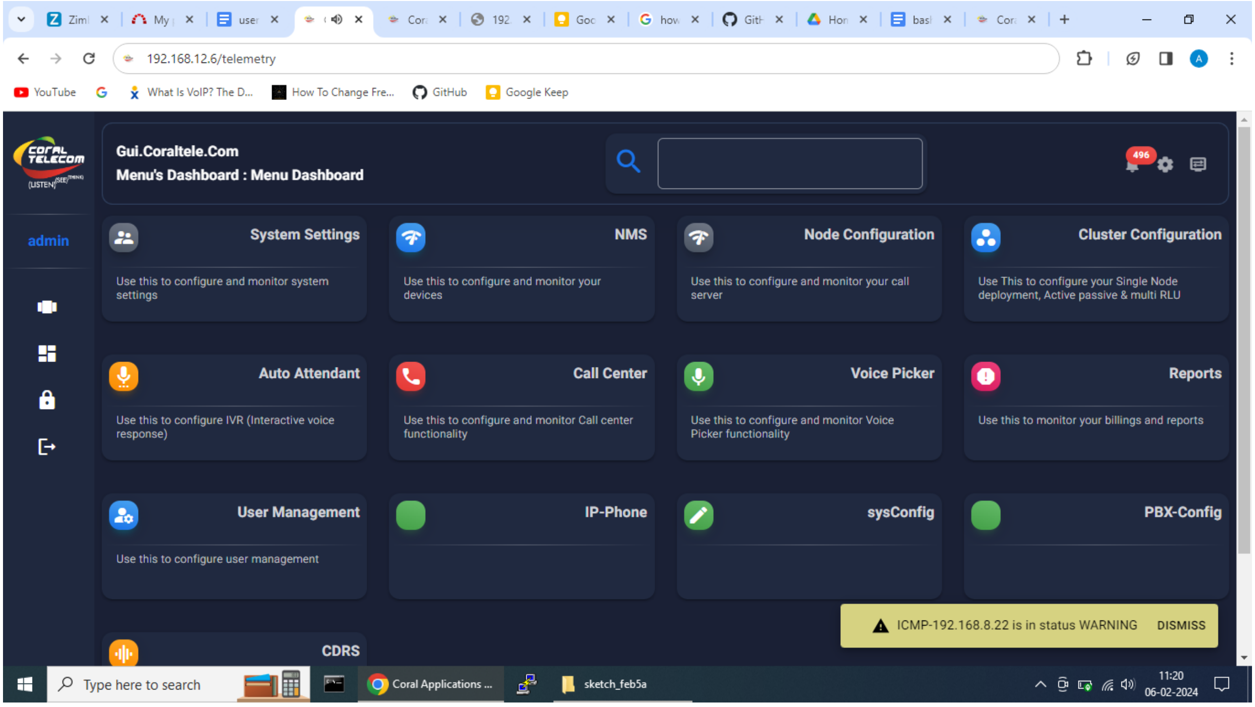
Password

LOGIN

ver 1.0.33

Username – Ask the Administrator for the username & password

When you log in you may see this type of pages:



Gui.Coraltele.Com

Menu's Dashboard : Menu Dashboard

System Settings – Use this to configure and monitor system settings

NMS – Use this to configure and monitor your devices

Node Configuration – Use this to configure and monitor your call server

Cluster Configuration – Use This to configure your Single Node deployment, Active passive & multi RLU

Auto Attendant – Use this to configure IVR (Interactive voice response)

Call Center – Use this to configure and monitor Call center functionality

Voice Picker – Use this to configure and monitor Voice Picker functionality

Reports – Use this to monitor your billings and reports

User Management – Use this to configure user management

IP-Phone

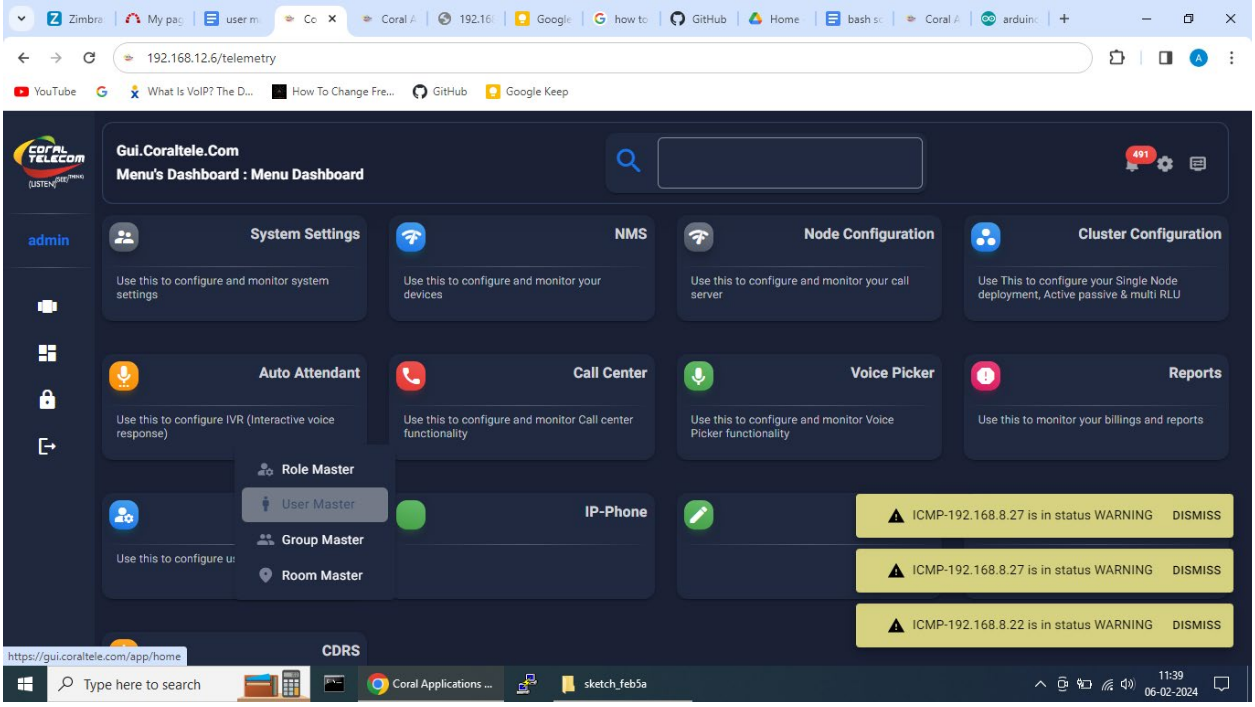
sysConfig

PBX-Config

CDRS

ICMP-192.168.8.22 is in status WARNING DISMISS

You want to left-click on “**user management**” You can see this type of option :



Gui.Coraltele.Com
Menu's Dashboard : Menu Dashboard

System Settings
Use this to configure and monitor system settings

NMS
Use this to configure and monitor your devices

Node Configuration
Use this to configure and monitor your call server

Cluster Configuration
Use This to configure your Single Node deployment, Active passive & multi RLU

Auto Attendant
Use this to configure IVR (Interactive voice response)

Call Center
Use this to configure and monitor Call center functionality

Voice Picker
Use this to configure and monitor Voice Picker functionality

Reports
Use this to monitor your billings and reports

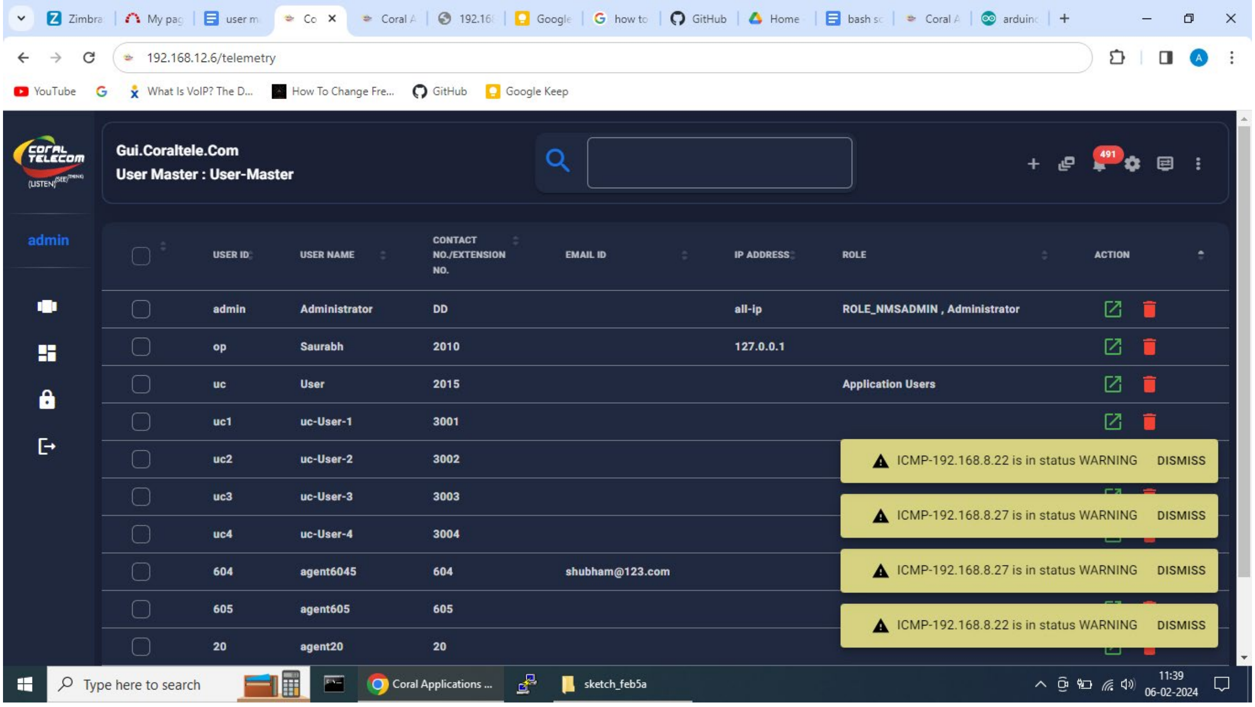
Use this to configure us

IP-Phone

CDRS

ICMP-192.168.8.27 is in status WARNING DISMISS
ICMP-192.168.8.27 is in status WARNING DISMISS
ICMP-192.168.8.22 is in status WARNING DISMISS

You want to select “**user master** “Now a page looks like this:

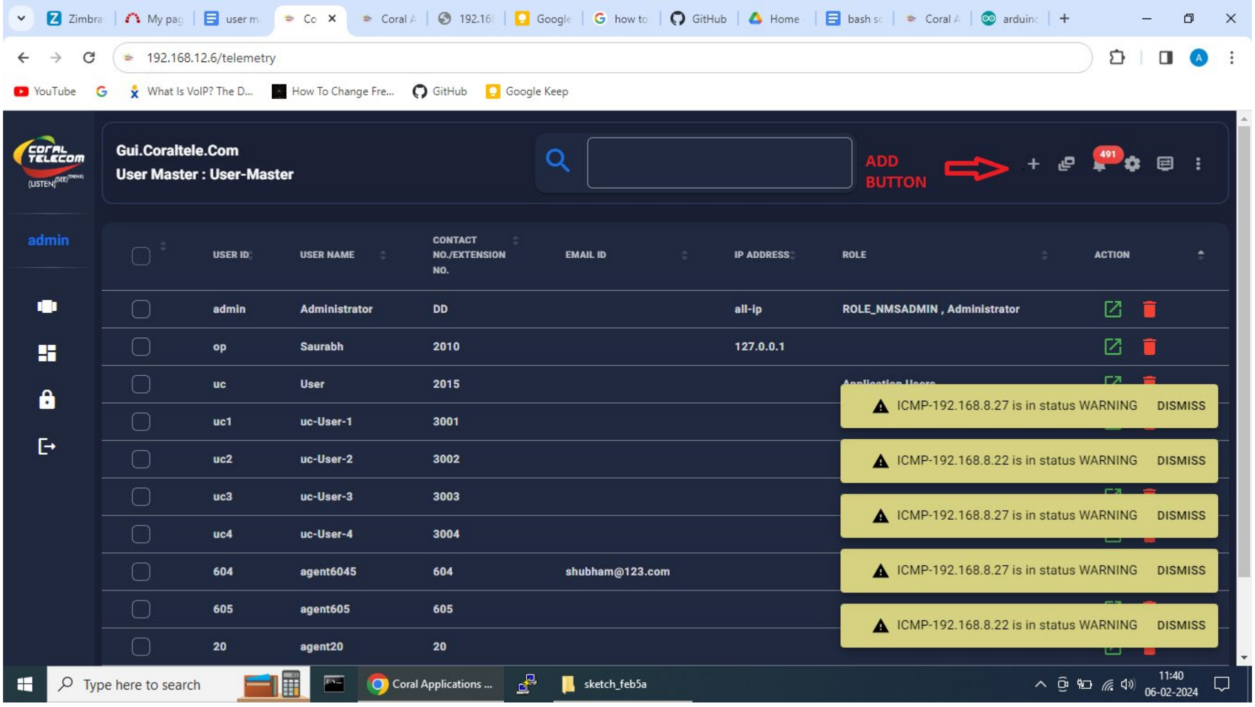


Gui.Coraltele.Com
User Master : User-Master

□	USER ID	USER NAME	CONTACT NO./EXTENSION NO.	EMAIL ID	IP ADDRESS	ROLE	ACTION
□	admin	Administrator	DD		all-ip	ROLE_NMSADMIN , Administrator	✎ 🗑
□	op	Saurabh	2010		127.0.0.1		✎ 🗑
□	uc	User	2015			Application Users	✎ 🗑
□	uc1	uc-User-1	3001				✎ 🗑
□	uc2	uc-User-2	3002				
□	uc3	uc-User-3	3003				
□	uc4	uc-User-4	3004				
□	604	agent6045	604	shubham@123.com			
□	605	agent605	605				
□	20	agent20	20				

ICMP-192.168.8.22 is in status WARNING DISMISS
ICMP-192.168.8.27 is in status WARNING DISMISS
ICMP-192.168.8.27 is in status WARNING DISMISS
ICMP-192.168.8.22 is in status WARNING DISMISS

Click on the add button and fill in the details (WHICH ARE HIGHLIGHTED WITH RED COLOUR)



	USER ID	USER NAME	CONTACT NO./EXTENSION NO.	EMAIL ID	IP ADDRESS	ROLE	ACTION
□	admin	Administrator	DD		all-ip	ROLE_NMSADMIN , Administrator	
□	op	Saurabh	2010		127.0.0.1		
□	uc	User	2015			Application Users	
□	uc1	uc-User-1	3001				
□	uc2	uc-User-2	3002				
□	uc3	uc-User-3	3003				
□	uc4	uc-User-4	3004				
□	604	agent6045	604	shubham@123.com			
□	605	agent605	605				
□	20	agent20	20				

ICMP-192.168.8.27 is in status WARNING DISMISS

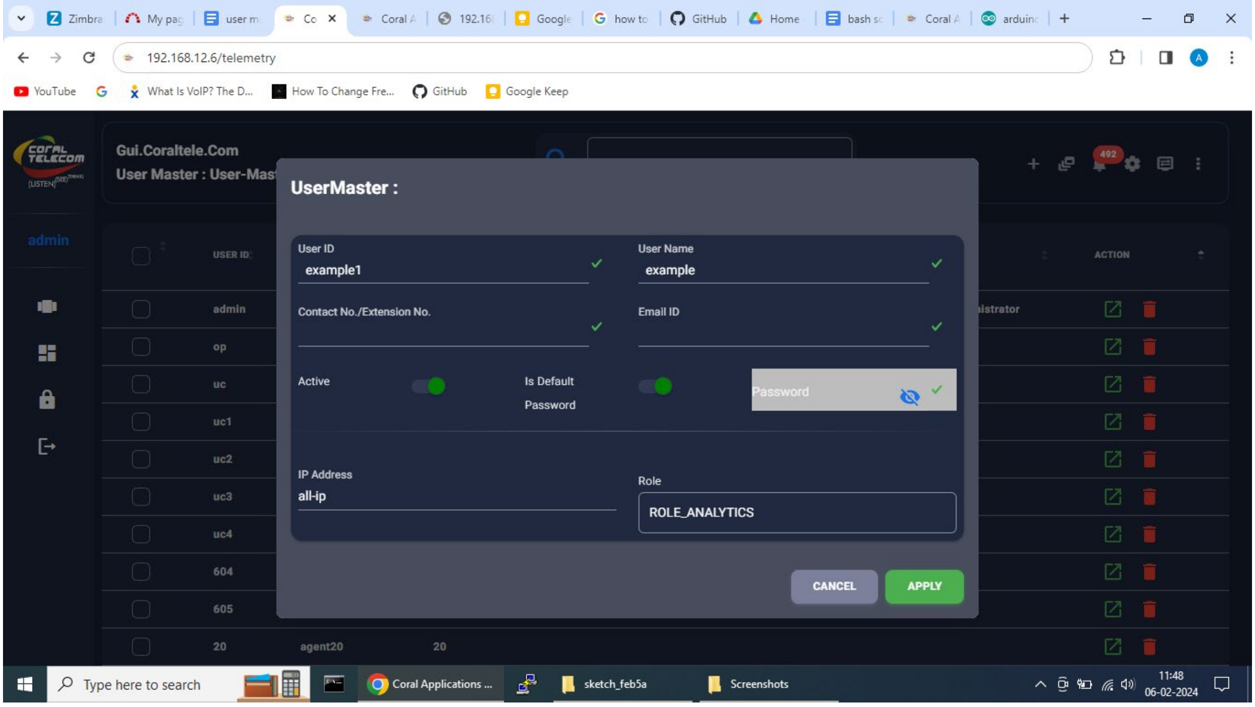
ICMP-192.168.8.22 is in status WARNING DISMISS

ICMP-192.168.8.27 is in status WARNING DISMISS

ICMP-192.168.8.27 is in status WARNING DISMISS

ICMP-192.168.8.22 is in status WARNING DISMISS

Then you see this type of page



UserMaster :

User ID: example1

User Name: example

Contact No./Extension No.:

Email ID:

Active ●

Is Default Password ●

Password

IP Address: all-ip

Role: ROLE_ANALYTICS

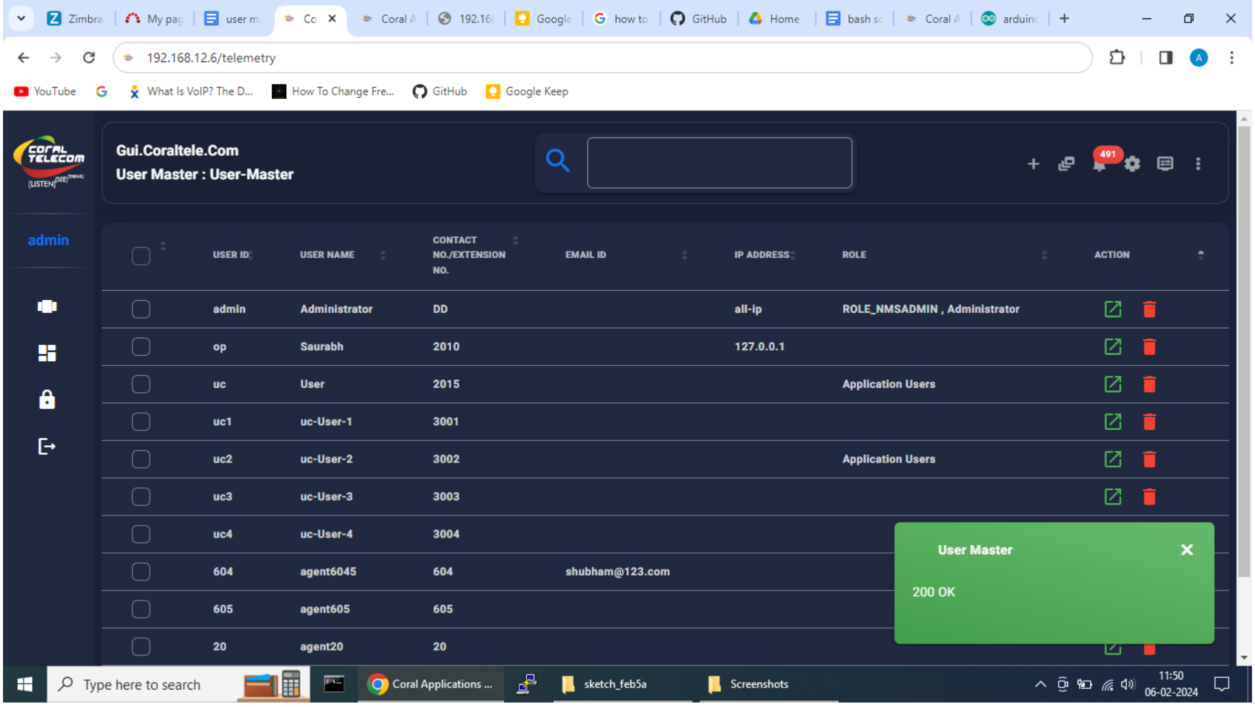
Here you want to add details

You have some optional details like contact no, email ID etc.

You want to select the role as “**ROLE_ANALYTICS**”

Then press apply

You received a message of 200 ok



Gui.Coraltele.Com
User Master : User-Master

admin

	USER ID	USER NAME	CONTACT NO./EXTENSION NO.	EMAIL ID	IP ADDRESS	ROLE	ACTION
□	admin	Administrator	DD		all-ip	ROLE_NMSADMIN , Administrator	
□	op	Saurabh	2010		127.0.0.1		
□	uc	User	2015			Application Users	
□	uc1	uc-User-1	3001				
□	uc2	uc-User-2	3002			Application Users	
□	uc3	uc-User-3	3003				
□	uc4	uc-User-4	3004				
□	604	agent6045	604	shubham@123.com			
□	605	agent605	605				
□	20	agent20	20				

User Master ✕

200 OK

Now your user is created ‘for example here username is **example1**’

And the password is a default which is “**12345**”

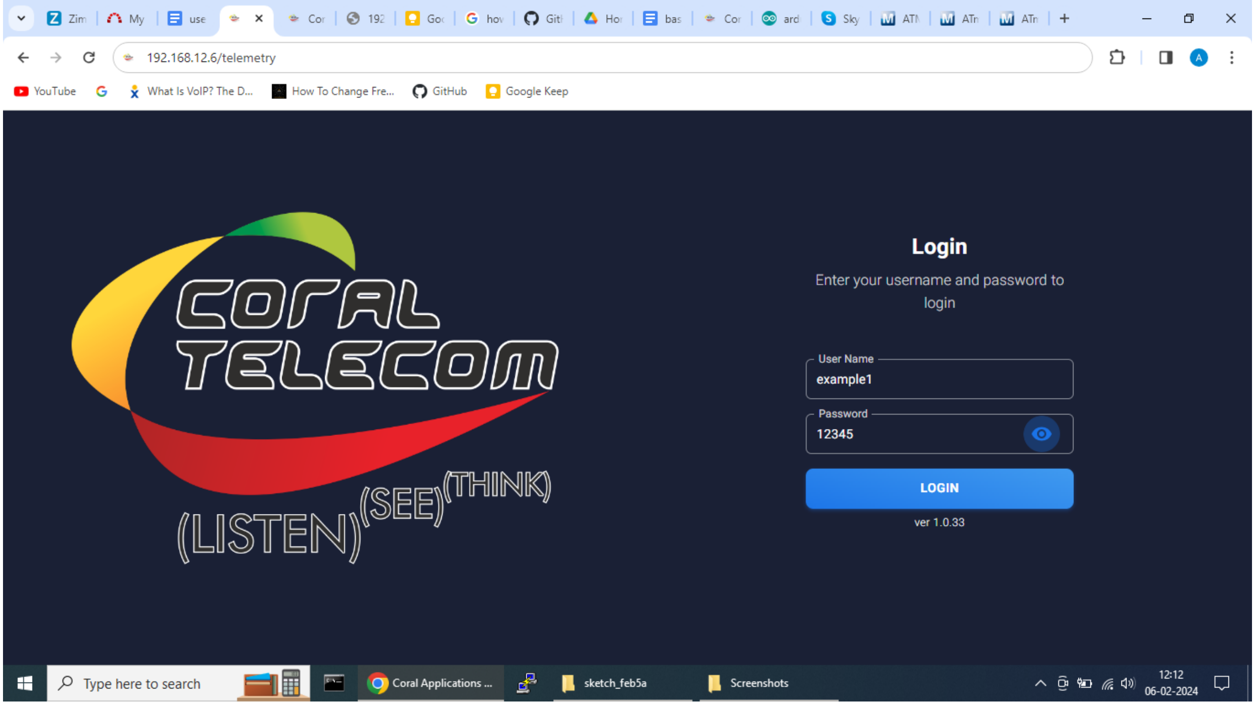
Step 3 - LOGIN WITH YOUR CREATED USER

Now login to NMS from your browser

NMS IP or domain - 192.168.12.6

Here login details are your created USER and default PASSWORD

For example: Username - example1 Password - 12345



Login

Enter your username and password to login

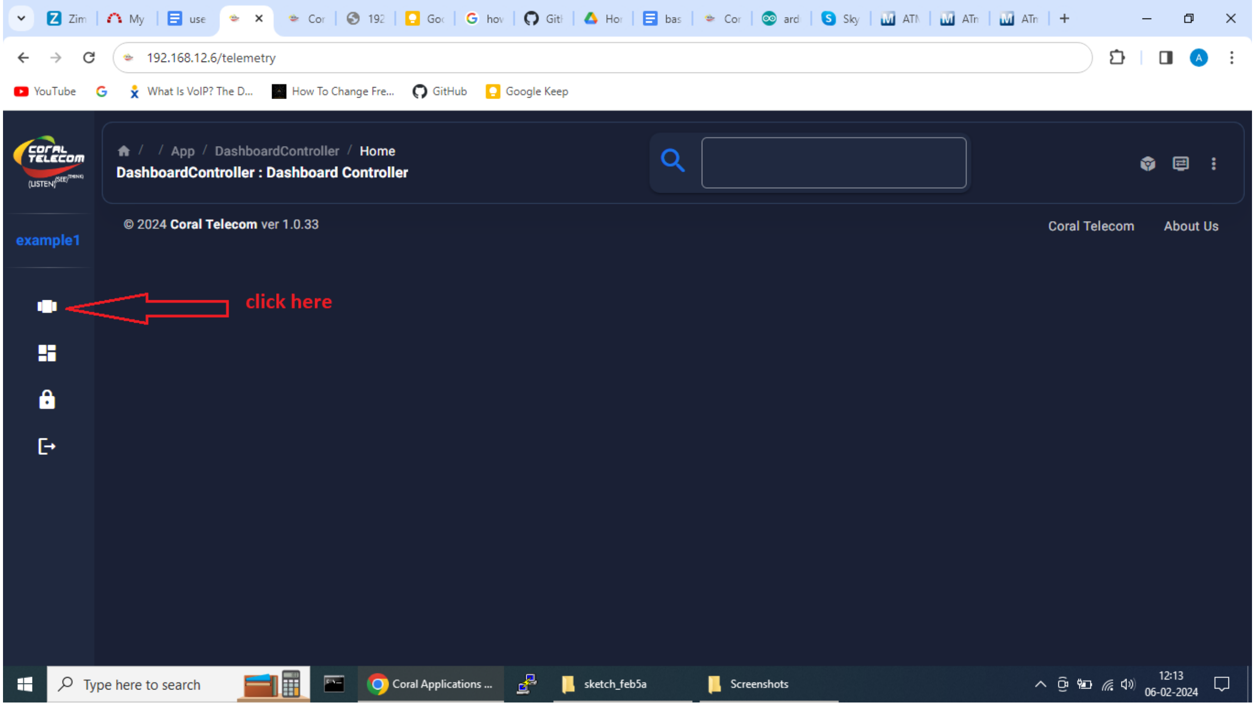
User Name: example1

Password: 12345

LOGIN

ver 1.0.33

After logging in you see this page and click on the arrow shown area:

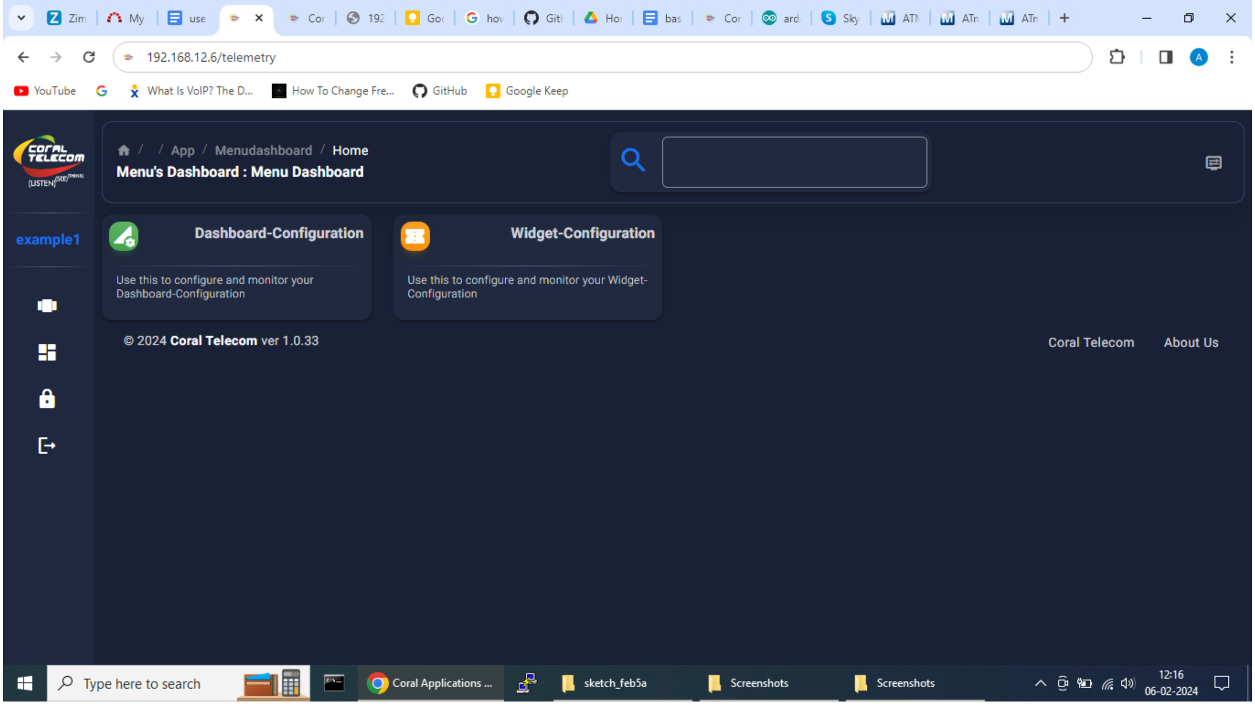


192.168.12.6/telemetry

DashboardController : Dashboard Controller

© 2024 Coral Telecom ver 1.0.33

STEP 4 - CREATE A DASHBOARD FOR YOUR SENSOR DATA



192.168.12.6/telemetry

Menu's Dashboard : Menu Dashboard

Dashboard-Configuration

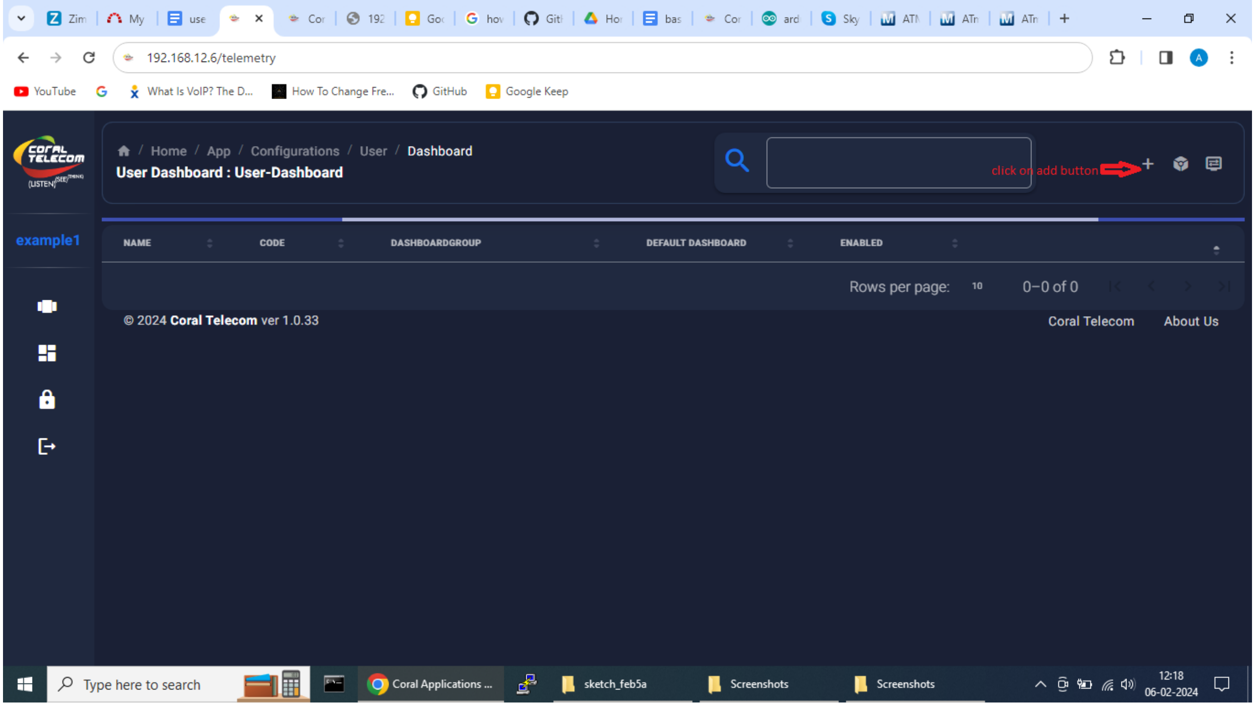
Use this to configure and monitor your Dashboard-Configuration

Widget-Configuration

Use this to configure and monitor your Widget-Configuration

© 2024 Coral Telecom ver 1.0.33

After clicking on "**Dashboard-configuration**" You will see this page, click on the add button



Home / App / Configurations / User / Dashboard

User Dashboard : User-Dashboard

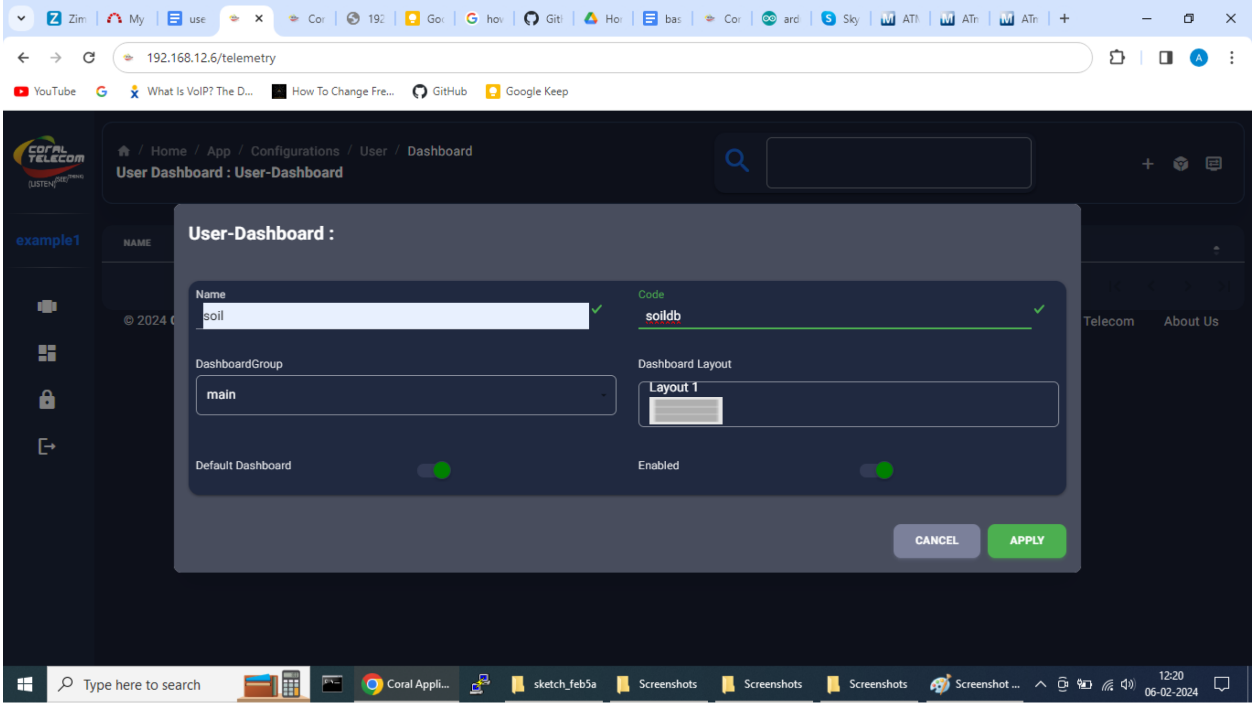
click on add button

NAME	CODE	DASHBOARDGROUP	DEFAULT DASHBOARD	ENABLED

Rows per page: 10 0–0 of 0

© 2024 Coral Telecom ver 1.0.33

You see this pop-up screen and you want to fill in the details for the sensor and you want to make the Dashboard group “**main**” (you can select it from drop-down in the dashboard group and click on the main option), enable/active all given option with your required Dashboard layout then click on apply



User-Dashboard :

Name: soil

Code: soildb

DashboardGroup: main

Dashboard Layout: Layout 1

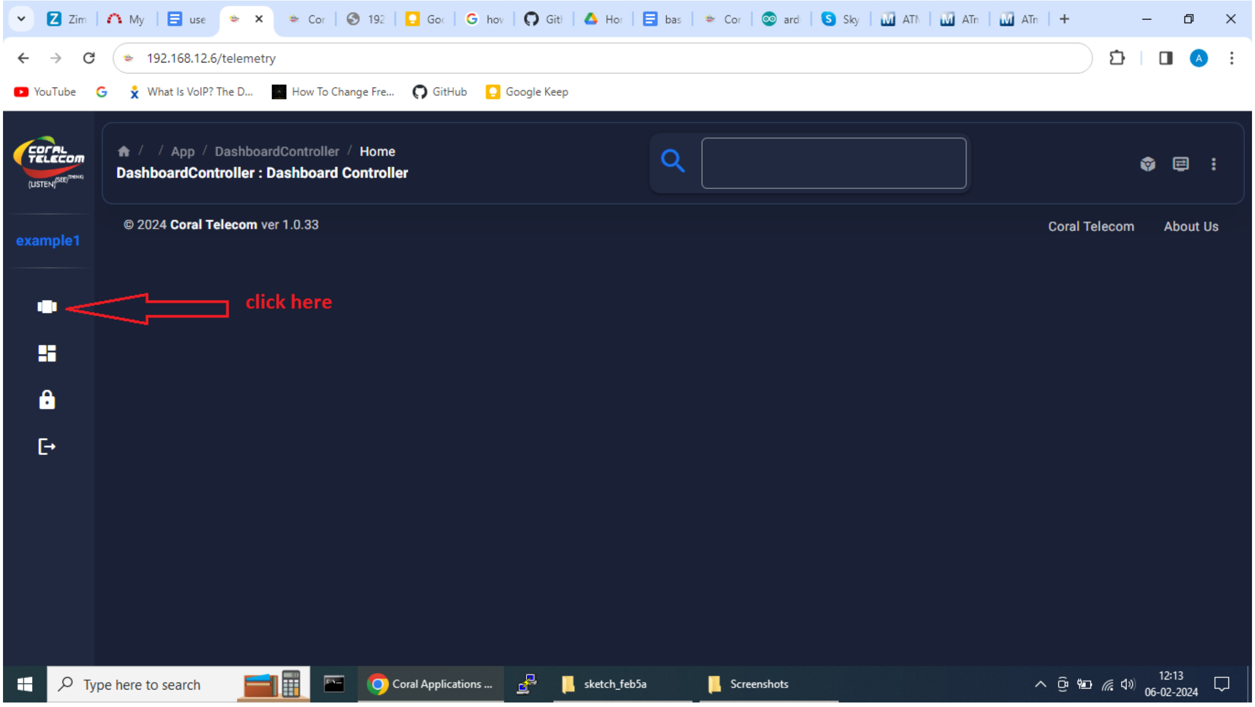
Default Dashboard

Enabled

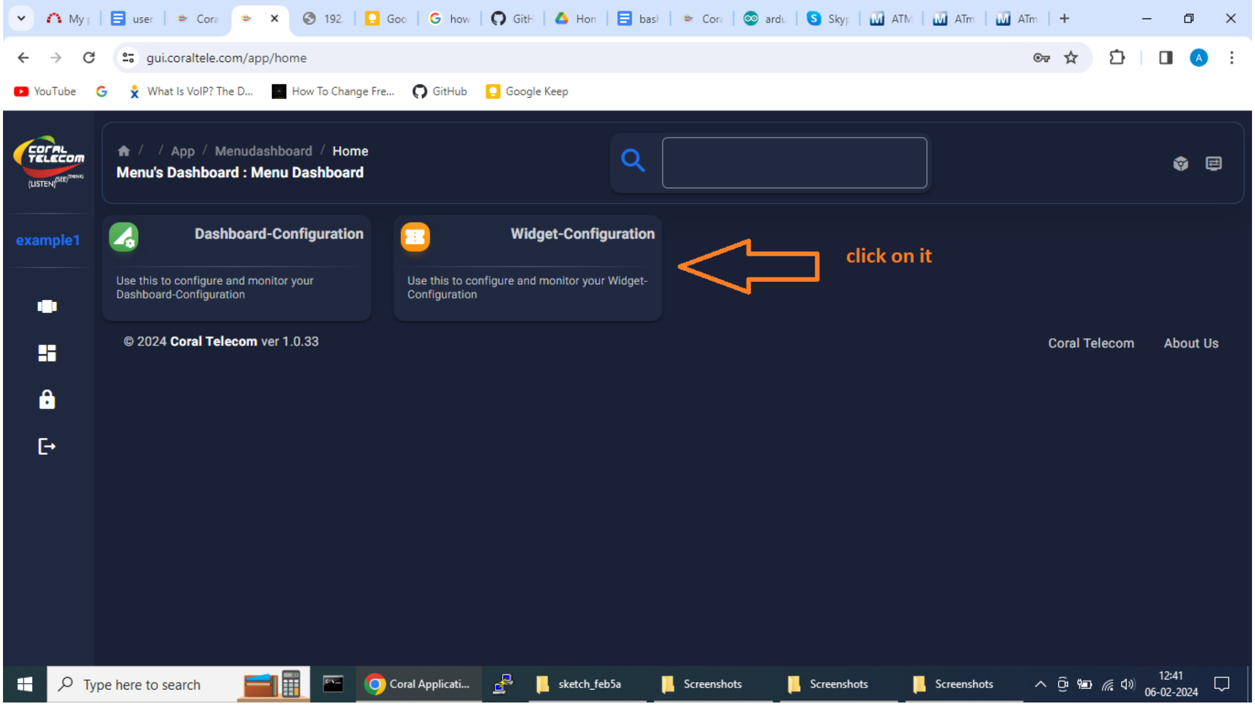
After clicking on apply you will see response message with an API KEY you does not work of API KEY.

STEP 5 - ADD WIDGET FOR SENSOR

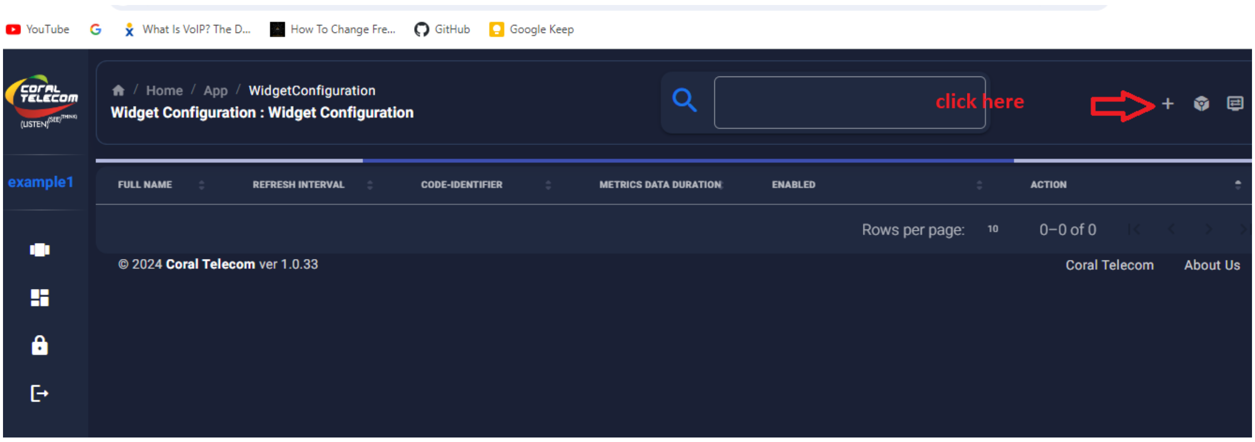
For click on this this button which is shown in picture


192.168.12.6/telemetry
DashboardController : Dashboard Controller
example1
© 2024 Coral Telecom ver 1.0.33
Coral Telecom
About Us
click here

Then click on Widget-configuration

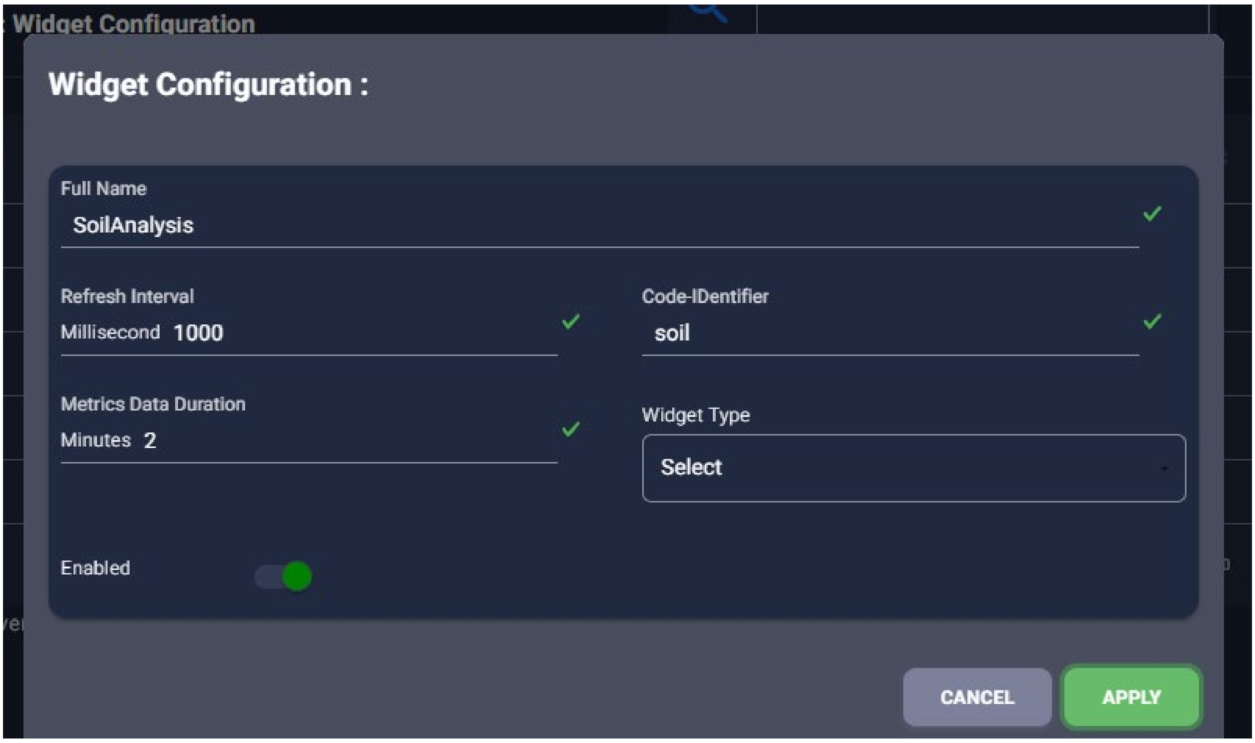

gui.coraltele.com/app/home
Menu's Dashboard : Menu Dashboard
example1
Dashboard-Configuration
Use this to configure and monitor your Dashboard-Configuration
Widget-Configuration
Use this to configure and monitor your Widget-Configuration
click on it
© 2024 Coral Telecom ver 1.0.33
Coral Telecom
About Us

Then click on add button to add widgets



Now you see a pop-up screen fill the sensor details and click on apply

[NOTE - Right side of the add button a wheel symbol is present, which is used to generate an authentication token it is unique, when you click on it a warning pop-up window will open now you have to option one for cancel and second agree by clicking on agree it generates a new authentication token]



Full Name: SoilAnalysis ✓

Refresh Interval: Millisecond 1000 ✓

Code-IDentifier: soil ✓

Metrics Data Duration: Minutes 2 ✓

Widget Type: Select

Enabled ●

Details of given tags

Full Name - write your sensor name

Refresh interval - interval to refresh the data of the sensor

Metrics data duration - the time limit shown in the graph of the sensor

Widget type - select the widget, you want like line chart, multi-series data etc.

After clicking on apply a message will pop up with the sensor ID on the right-side corner of your screen from there copy the sensor ID.

If you want to add multiple, then follow STEP 5.

You need to paste your copied sensor ID to the file whose name is a sensor, which is placed on

Path - /etc/defaults/sensor

In this file replace your sensor ID

For edit in this file, you want to log on to Coral Gyan using the putty application with user, password and IP which is written below

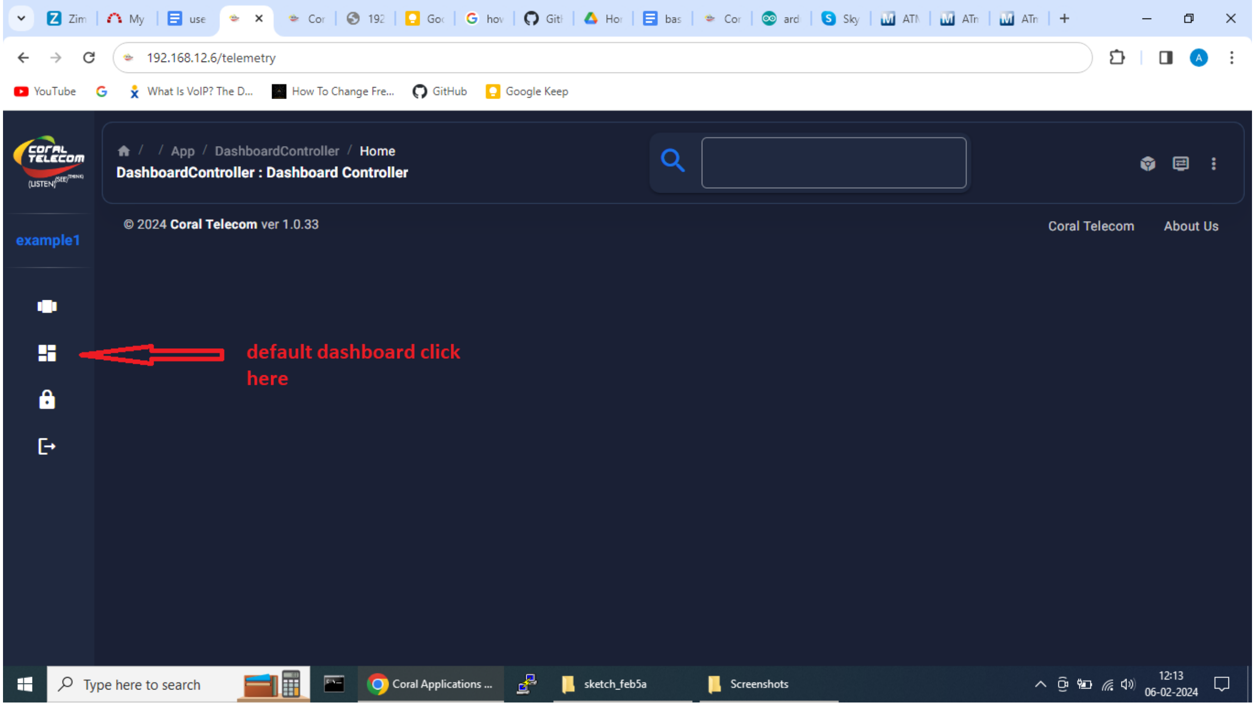
IP address 192.168.12.7 or 192.168.12.8

Username - ****

Password - **********

Now after replacing the sensor key, you want to login to your created user and click on this

Symbol or default dashboard



192.168.12.6/telemetry

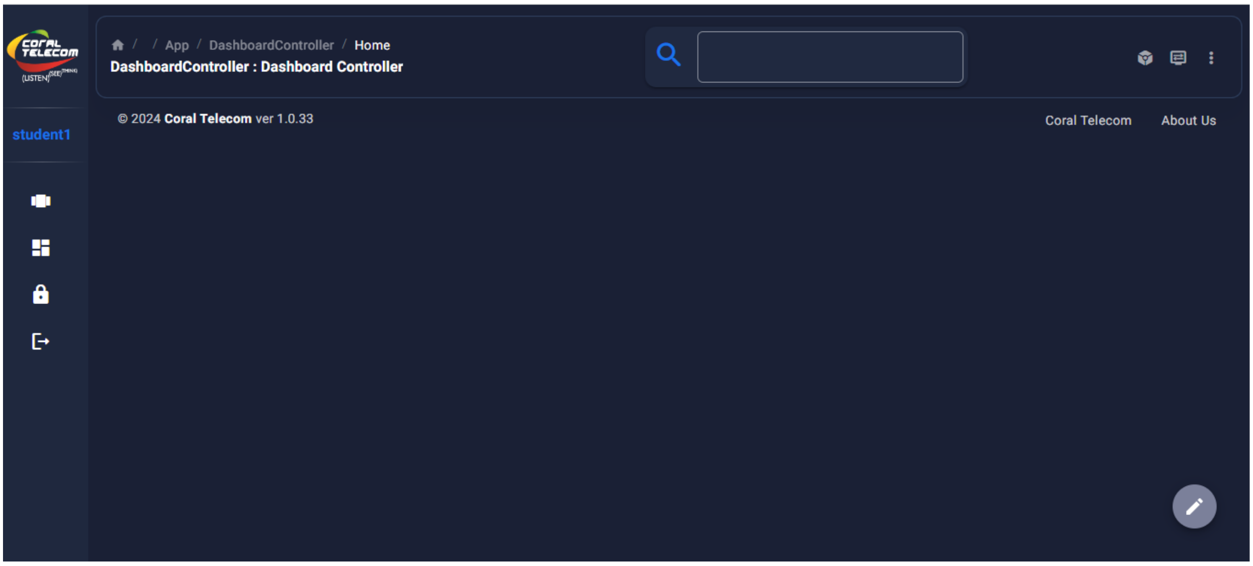
App / DashboardController / Home

DashboardController : Dashboard Controller

© 2024 Coral Telecom ver 1.0.33

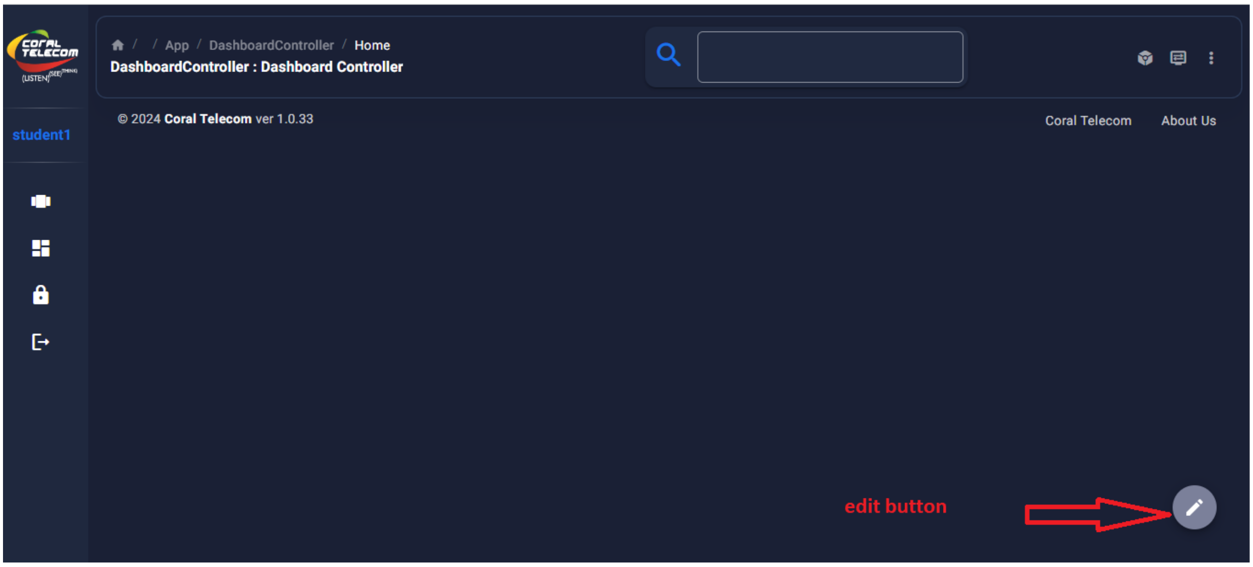
default dashboard click here

Then you see a screen like this

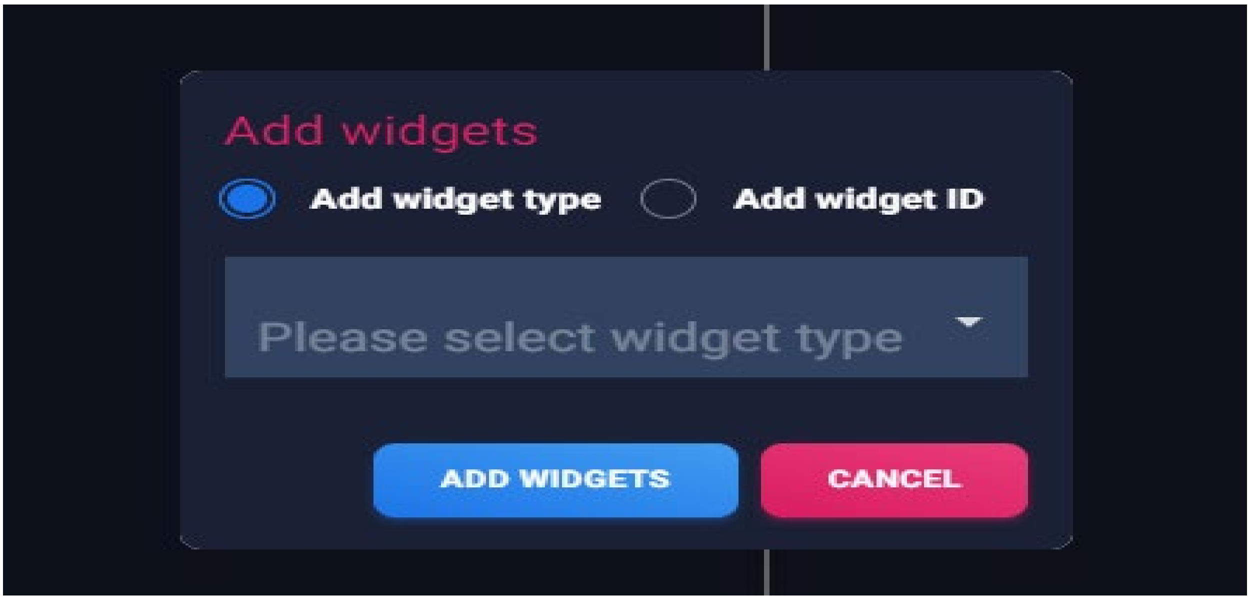

/ / App / DashboardController / Home
DashboardController : Dashboard Controller
student1
© 2024 Coral Telecom ver 1.0.33
Coral Telecom
About Us

STEP 6 - ADD WIDGET GRAPH TO YOUR DEFAULT PAGE

You want to click on edit button to add widget

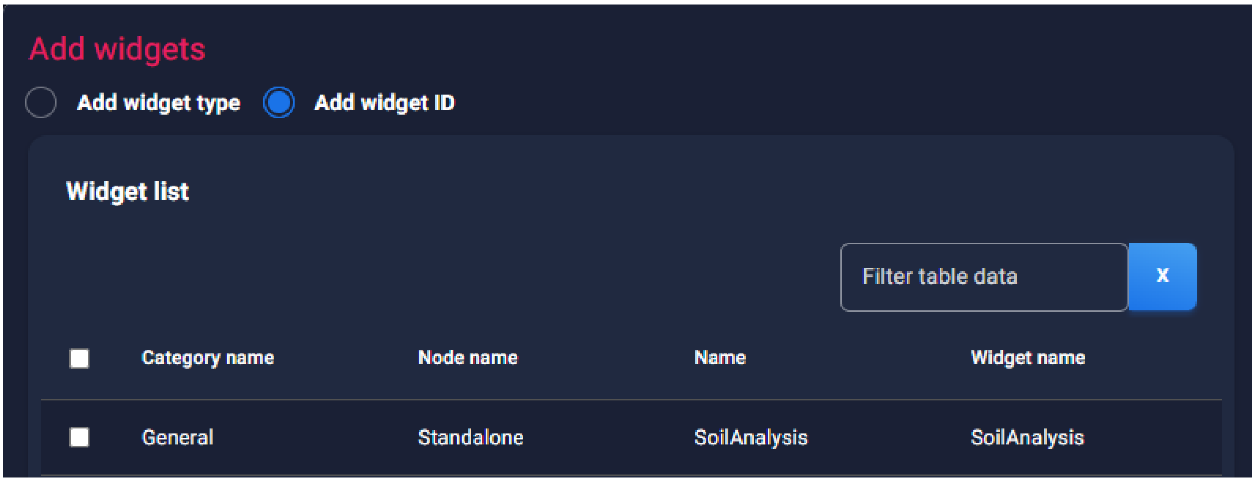

/ / App / DashboardController / Home
DashboardController : Dashboard Controller
student1
© 2024 Coral Telecom ver 1.0.33
Coral Telecom
About Us
edit button

The pop-up will open and show this type of option.



Select the "**Add Widget ID** "

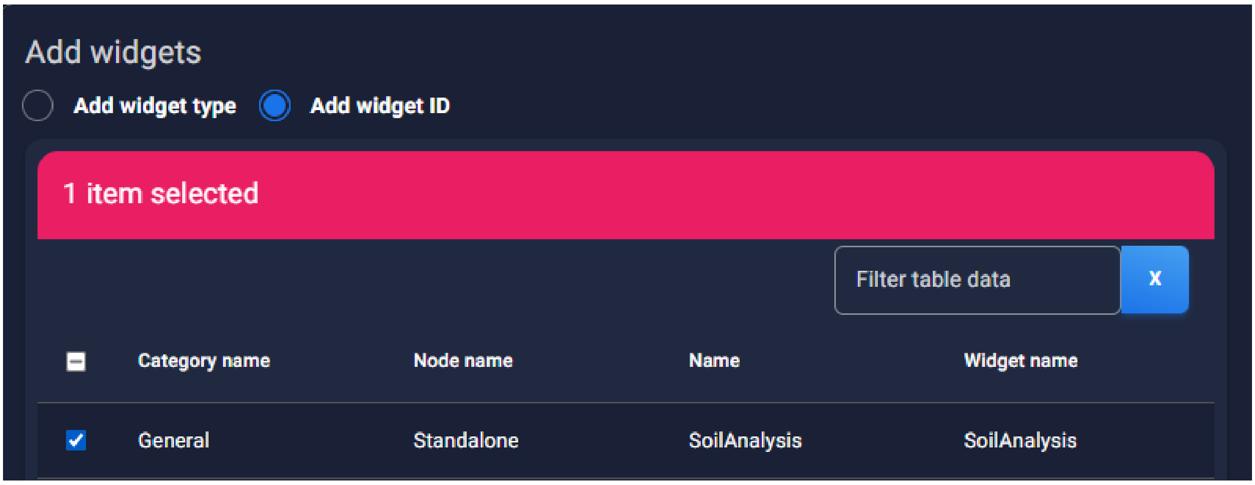
And you will see list of your widgets like this



Widget list

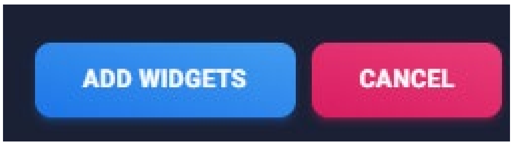
□	Category name	Node name	Name	Widget name
□	General	Standalone	SoilAnalysis	SoilAnalysis

Select your widget and click on add widget button

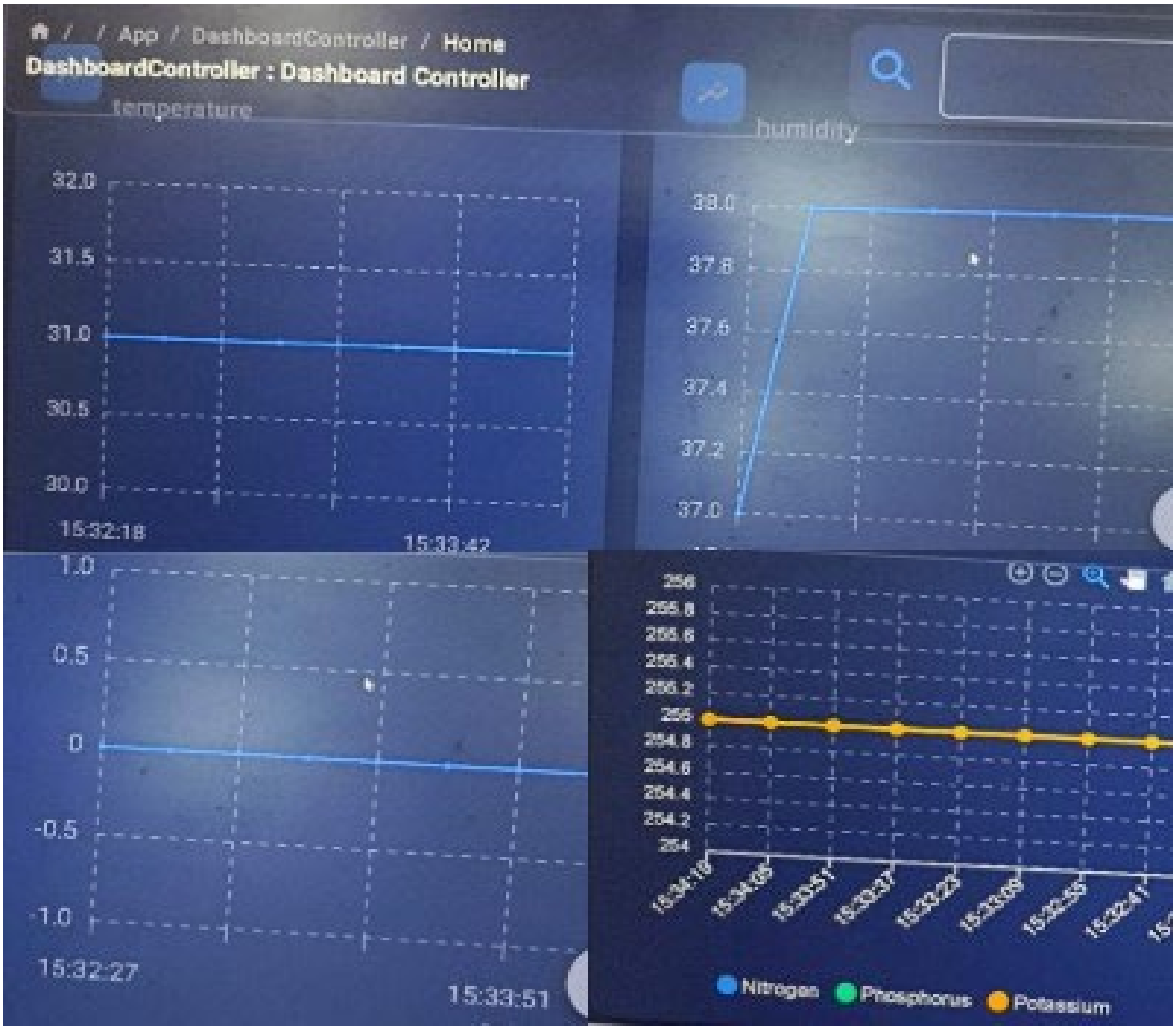


1 item selected

⊟	Category name	Node name	Name	Widget name
☑	General	Standalone	SoilAnalysis	SoilAnalysis



Now your data is shown on the dashboard



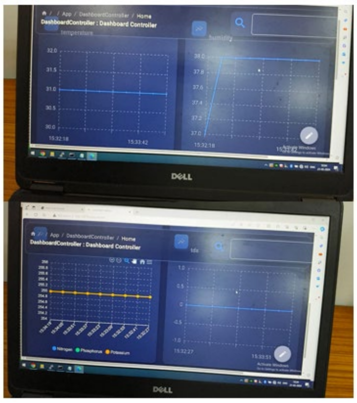
If you add multiple widgets for multiple sensors you want to redo **Step 6**.

Coral Gyan Use Case With 5G Evaluation Board (Anubhav)

Overview

Coral Gyan transmits sensor data via a 5G Evaluation Board to the NMS (Network Management System) over a 5G network.

Wireless connectivity is essential for data transmission.



Requirements

Accessing Coral Gyan

- ✓ IP Address, Username(gyan),
- ✓ and Password (CgYaN@817&&) required.
- ✓ Ensure Coral Gyan has a Wi-Fi module or use a wireless dongle.

Configuring Wireless LAN

Turn on Coral Gyan and access terminal.

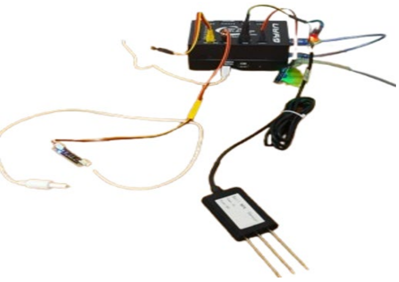
Run the command:

```
raspi-config
```

Navigate to System Options → Wireless LAN.

Enter Wi-Fi SSID and Password for network connection.

Data Transmission Process



1. Connect Coral Gyan to Wifi using the configuration steps.
2. Sensor data is collected from connected devices.
3. 5G Evaluation Board transmits data securely over the 5G network.
4. Data reaches NMS for monitoring and analysis.

Key Considerations & Troubleshooting

- Check Wifi Connection – Ensure SSID (QSoftAP) & Password (1234567890) are correct.
- Verify 5G Evaluation Board Setup – Proper integration with Coral Gyan.
- Monitor Data Transfer – Use logs to verify transmission to NMS.

Conclusion

- Efficient data transmission using Coral Gyan + 5G Evaluation Board.
- Real-time monitoring and analytics via 5G network.
- Scalable for IoT applications and edge computing.

IoT Use-Cases and Deployment Scenarios across key Sectors

Smart Cities: IoT infrastructure integrates smart lighting systems, adaptive traffic control, waste management automation, and surveillance networks. 5G-enabled edge devices such as AI-powered cameras and multi-sensor IoT boxes facilitate real-time analytics for public safety and environmental monitoring.

Agriculture (Precision Farming): IoT-based solutions enable granular control over irrigation, soil nutrient profiling, and livestock geofencing. Devices like the Coral Gyan Sensor Box collect and transmit soil moisture, pH, and ambient data to cloud platforms for actionable insights.

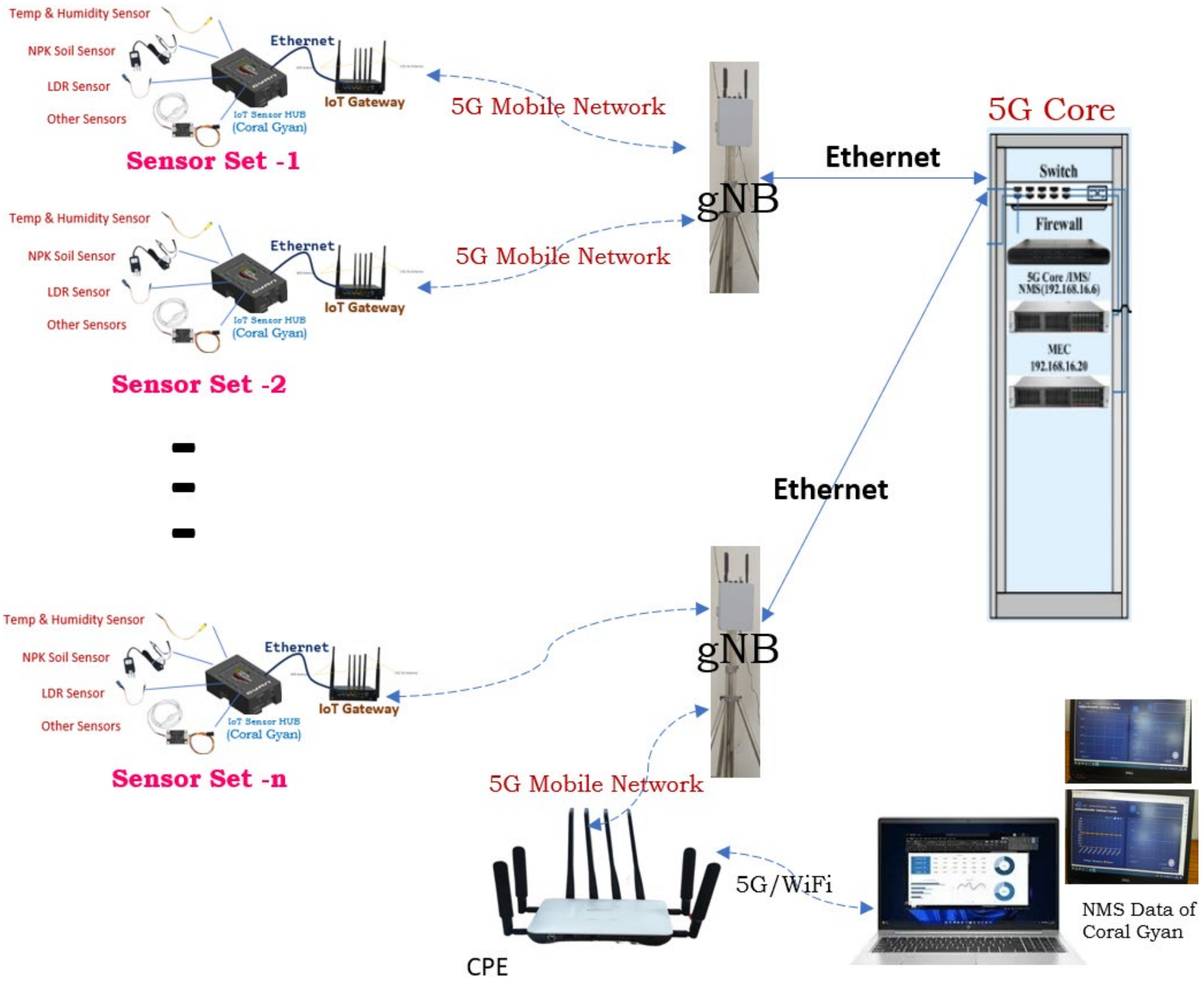
Smart Homes: Home automation ecosystems leverage IoT protocols (e.g., Zigbee, Z-Wave, MQTT) to manage lighting, HVAC, and security systems. AI-integrated 5G surveillance cameras provide low-latency, high-resolution monitoring with edge-based anomaly detection.

Automotive & Transportation: Connected vehicle platforms utilize IoT modules for V2X communication, fleet telemetry, and predictive route optimization. Real-time data exchange enhances traffic flow, fuel efficiency, and driver safety.

Industrial IoT (IIoT): Manufacturing units deploy IoT sensors for equipment health monitoring, energy usage tracking, and process automation. Integration with SCADA systems and predictive analytics engines enables proactive maintenance and operational efficiency.

Healthcare: IoT-enabled medical devices support remote patient monitoring, biometric data logging, and AI-assisted diagnostics. Wearables and smart implants transmit real-time vitals to healthcare providers, enabling faster response and personalized care.

An example set up for utilizing the equipment in the 5G Lab for agriculture sector is as follows:



Data Flow Example

For real world application, use the SIM of available 5G Mobile of a TSP in the IoT gateway and CPE, also configure the destination cloud address for data storing in the Coral Gyan configuration. Now utilize these real time data to actuate various processes like irrigation system, etc.

Utilize Artificial Intelligence (AI) and Machine Learning (ML) to make intelligent decisions. to enhance automation, optimize decision-making, and drive innovation.

Communication Flow:

Step-by-Step Communication Flow:

i. Coral Gyan (IoT Sensor Box) Data Collection & Transmission

The Coral Gyan IoT Sensor Box collects sensor data
It sends this data to the IoT Gateway over Ethernet.

ii. IoT Gateway to 5G Radio Transmission

The IoT Gateway acts as an intermediary between the sensor box and the 5G network.
It aggregates the sensor data and forwards it to the 5G Radio when it detects an active 5G connection via 5G SIM.

iii. 5G Radio to 5G Core Transmission

The 5G Radio (gNodeB) receives the IoT data from the IoT Gateway.
It then transmits the data to the 5G Core.
The 5G Core handles authentication, routing.

iv. 5G Core to NMS Data Forwarding

The 5G Core forwards the received sensor data to the NMS (Network Management System).
The NMS is hosted on a cloud or an on-premises server and processes, stores, and visualizes the data.
The data can now be monitored through an NMS Dashboard.

Reverse Communication Flow:

i. 5G Radio Connection to 5G CPE

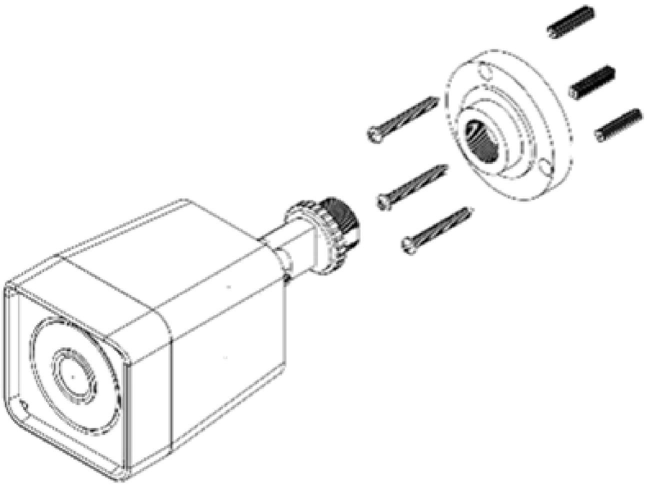
When a 5G CPE (Customer Premises Equipment) device is connected to the 5G Radio via a 5G SIM, it acts as a bridge between the 5G network and local devices (PCs, IoT devices, etc.).
The 5G CPE receives data from the 5G Radio and establishes an IP network for connected devices.

ii. 5G CPE to Computer System Transmission

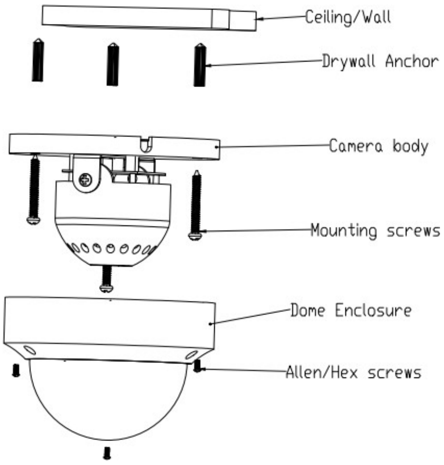
Any computer system, IoT device, or local network connected to the 5G CPE can now access the data from the NMS Dashboard.
The data can be accessed via a web interface.

NOTE: By default, the IP of the Coral Gyan is set to 192.168.11.35 but for using it for other devices you can use by setting it and DHCP or Static

5G AI Camera



Bullet Camera

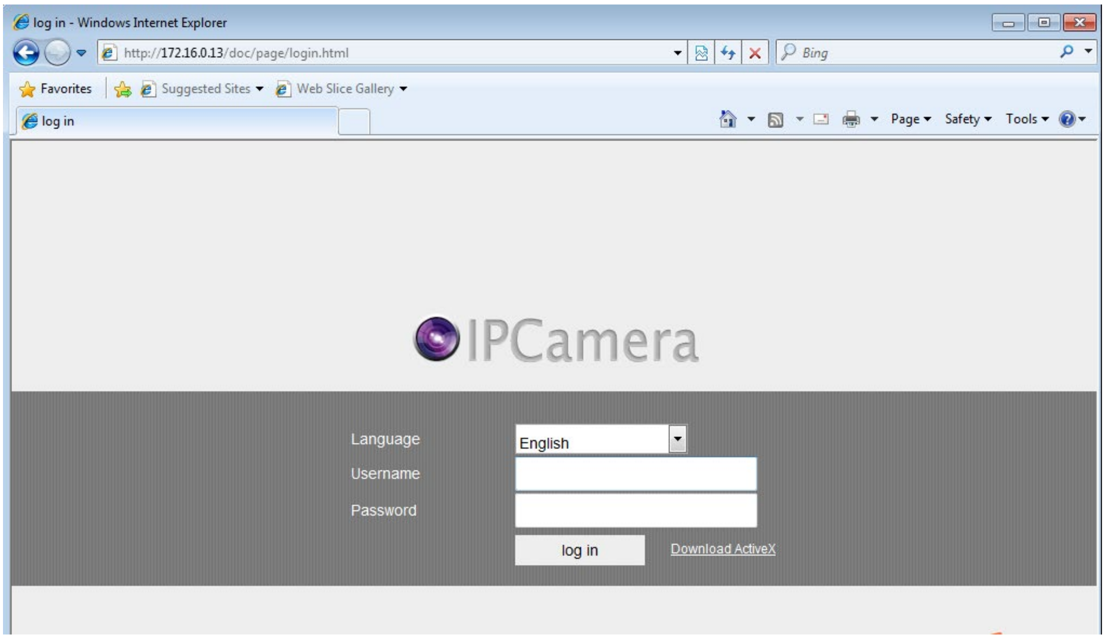


Dome Camera

Web Client User Manual

Install Active-x

Input username and password to log in (When first access, need to download the Active-X) :

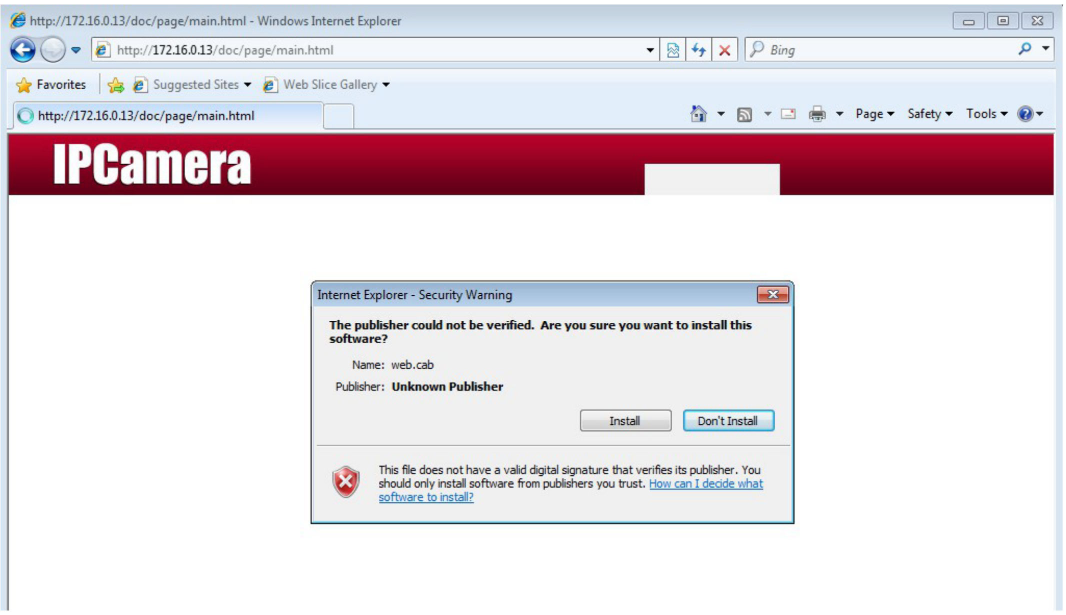


Language English

Username

Password

Choose a language Input username and password to log in, automatically prompt to install Active-x, Click “Install” to complete the Active-x installation.


http://172.16.0.13/doc/page/main.html - Windows Internet Explorer
http://172.16.0.13/doc/page/main.html
Bing
Favorites
Suggested Sites
Web Slice Gallery
http://172.16.0.13/doc/page/main.html
Page
Safety
Tools
IPCamera
Internet Explorer - Security Warning
Install
Don't Install

The publisher could not be verified. Are you sure you want to install this software?

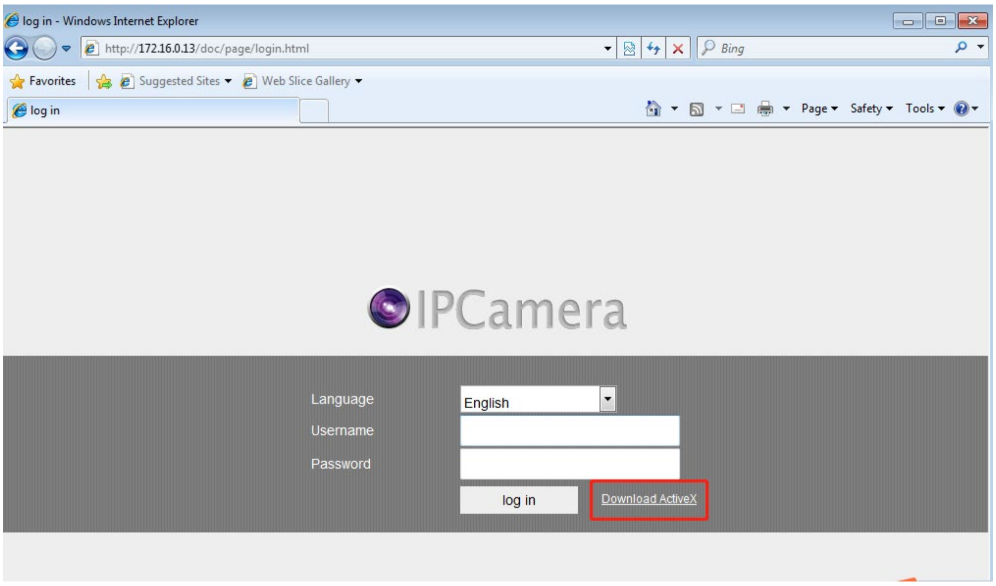
Name: web.cab

Publisher: **Unknown Publisher**

This file does not have a valid digital signature that verifies its publisher. You should only install software from publishers you trust. How can I decide what software to install?

When installation finished, there will appear a file folder named Web client in C:\Program Files \IPC Web. It is the downloaded file. If you delete the file, it will result in inablilty to log in the Web client. You should download again.

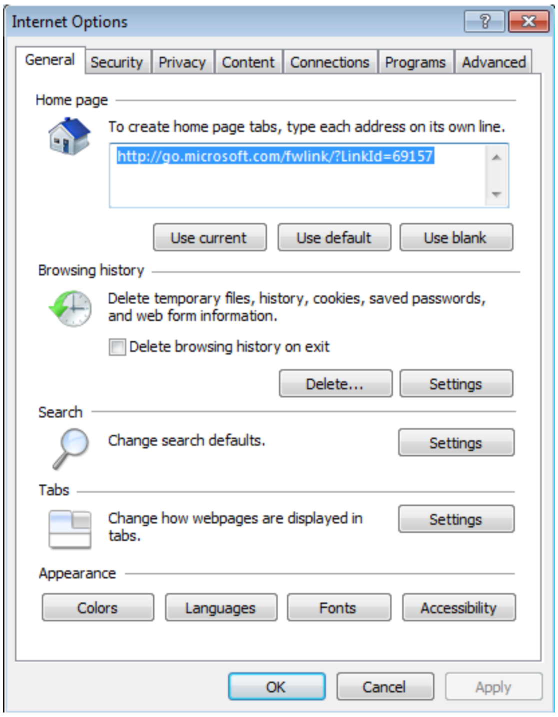
If you are not prompted to install the control, click " Download Active X"



Language: English
Username:
Password:

The file will be downloaded in C:\Users\Administrator\Downloads \TSIPCWebClient_EN_SetUp.exe, Double clicks to run this program.

FA Question : Why I cannot install the control?
Answer: Your IE browser intercept the installation; you need change the settings in IE Internet. choose the security option.



Home page

To create home page tabs, type each address on its own line.

http://go.microsoft.com/fwlink/?LinkId=69157

Browsing history

Delete temporary files, history, cookies, saved passwords, and web form information.

□ Delete browsing history on exit

Search

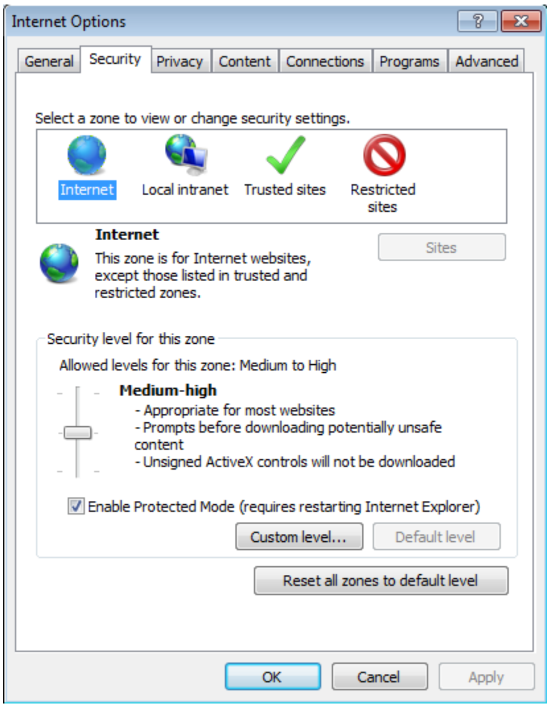
Change search defaults.

Tabs

Change how webpages are displayed in tabs.

Appearance

Choose “Security”



Select a zone to view or change security settings.

Internet

This zone is for Internet websites, except those listed in trusted and restricted zones.

Security level for this zone

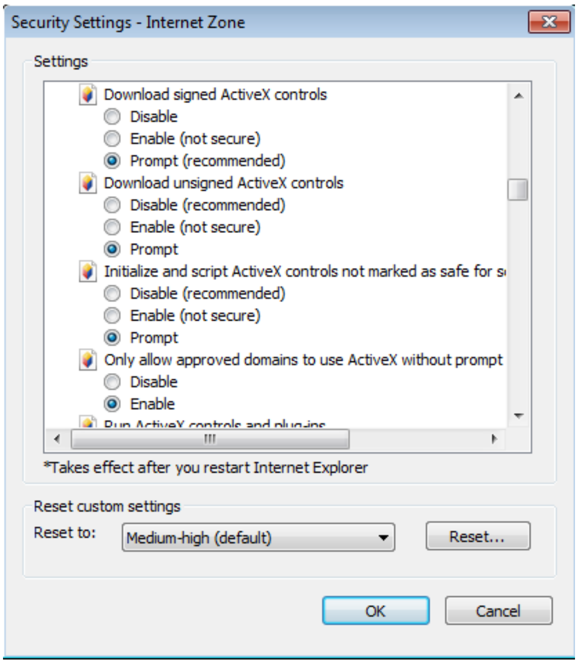
Allowed levels for this zone: Medium to High

Medium-high

- Appropriate for most websites
- Prompts before downloading potentially unsafe content
- Unsigned ActiveX controls will not be downloaded

☑ Enable Protected Mode (requires restarting Internet Explorer)

Choose “Custom Level” , Enable “Download unsigned Active X controls.”



Download signed ActiveX controls
○ Disable
○ Enable (not secure)
● Prompt (recommended)

Download unsigned ActiveX controls
○ Disable (recommended)
○ Enable (not secure)
● Prompt

Initialize and script ActiveX controls not marked as safe for s
○ Disable (recommended)
○ Enable (not secure)
● Prompt

Only allow approved domains to use ActiveX without prompt
○ Disable
● Enable

Run ActiveX controls and plug-ins

*Takes effect after you restart Internet Explorer

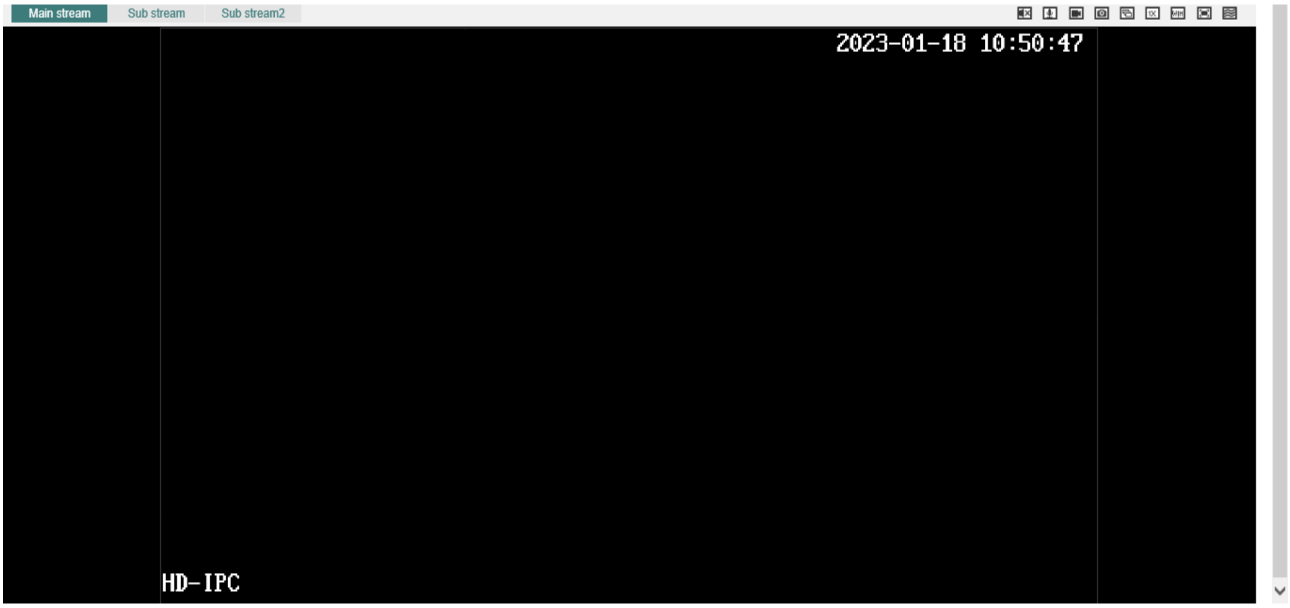
Reset to: Medium-high (default)

Then Re-enter the IP address of device, choose “install Active-X controls”.
Note: enter the username and password of device click log in to access web client.

FA Question: I have already downloaded the controls, why I still cannot log in Web client?
Answer: Please check if the network is normal or if the IP conflict with another device.

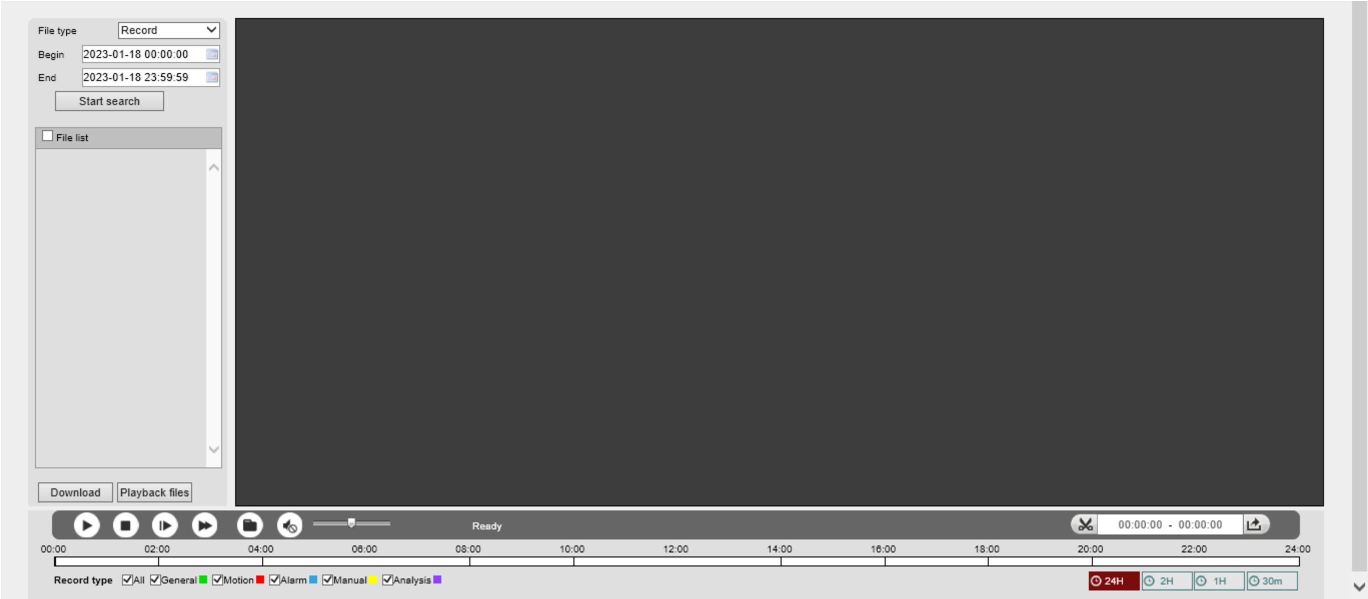
Web client operation

Preview interface :



Preview interface	Main stream | Sub stream | Sub stream2	Select Mainstream or Sub stream or Sub stream 2 to play video
	🔇	Input on the device, turn on the switch, you can hear the sound
	🎤	Turn on the switch, the device and the web can talk
	🎥	Record: Record local recordings and save them in C:\Record
	📷	Capture the preview screen and save it in C:\Picture
	⧉	Continuously capture three pictures and save them in C:\Picture
	1X	Original resolution display
	W|H	Suitable for display resolution
	⛶	Full-screen display
	≋ ─●── 3	fluency

Playback interface :

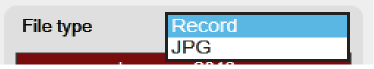
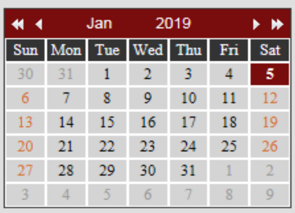
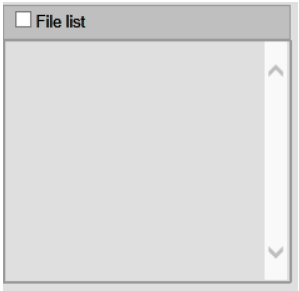
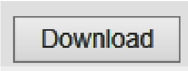
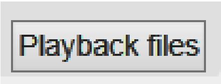
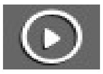
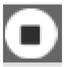
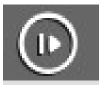


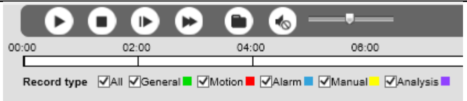
File type: Record

Begin: 2023-01-18 00:00:00

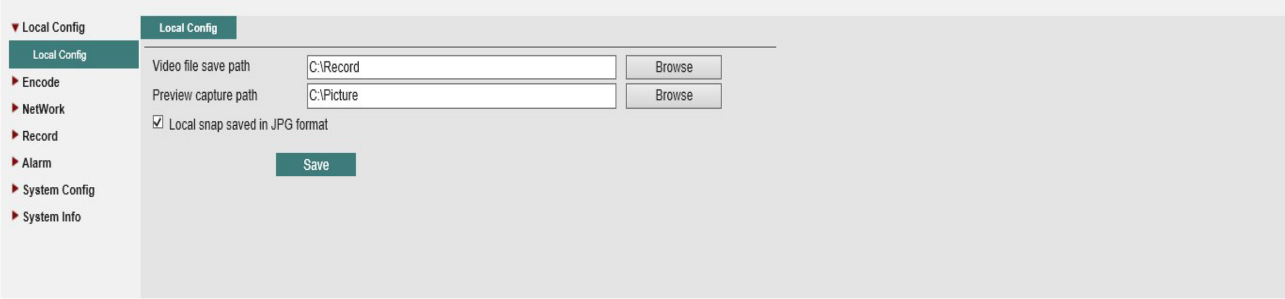
End: 2023-01-18 23:59:59

Record type: ☑All ☑General 🟩 ☑Motion 🟥 ☑Alarm 🟦 ☑Manual 🟨 ☑Analysis 🟪

Playback interface	 File type: Record JPG	Select the type of playback file
	 Jan 2019 Sun Mon Tue Wed Thu Fri Sat 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9	Select the playback date, The default is the current date
		Display videos or pictures saved in the TF card
		Download the video file to a local folder
		Play the video files
		Start playback
		Stop playback
		Slow playback

	⏩	Fast playback
	📂	Play local video files
		Sound Volume Up>Down and Mute
	 00:00 02:00 04:00 06:00 Record type ☑All ☑General 🟩 ☑Motion 🟥 ☑Alarm 🟦 ☑Manual 🟨 ☑Analysis 🟪	Show different recording categories

Configuration interface ：



▾ Local Config
Local Config
▸ Encode
▸ NetWork
▸ Record
▸ Alarm
▸ System Config
▸ System Info

Local Config

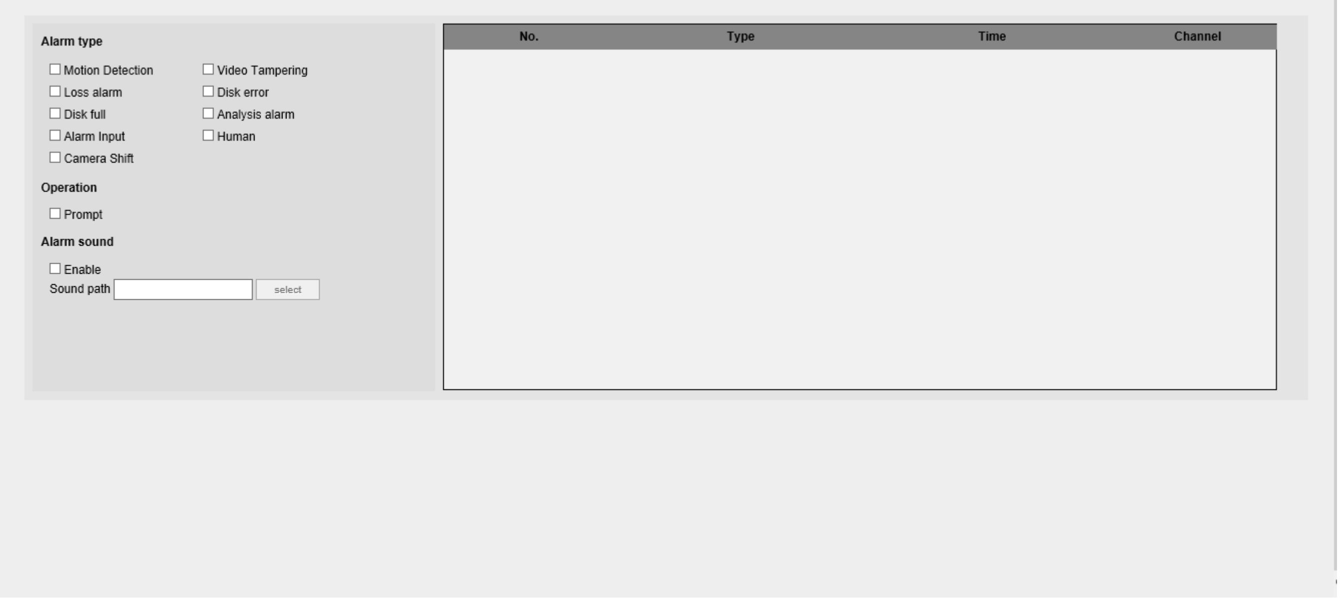
Video file save path: C:\Record [Browse]
Preview capture path: C:\Picture [Browse]
☑ Local snap saved in JPG format
[Save]

Local Config	Set the save path of the video and capture file. Capture file format (bmp, jpg)
Camera - Encode Config - Encode setting	Set camera encoding, resolution, frame rate, code stream and other parameters
Camera - Encode Config - Audio	Set the Encode mode, Sampling rate, Out volume and In volume.
Encode Config -Prev. Mode - Output model	Set display time, channel title and week title.
Encode Config -Prev. Mode- Display config	Set image properties, camera parameters, ISP parameters
Encode Config -Prev. Mode-	Set the profile, privacy masking and overlay
Network Config -TCP/IP	Set the device's IP, port, etc.
Network Config -Net service	Set Email, DDNS, RTSP, FTP, UPnP, P2P, IP filter, SNMP and HTTPS.
Record-Record-Record Plan	Length, Pre Record, Redundancy and Record Mode
Record-Record-Storage	Storage Device list, Set readable/writable, Set read only, Set Redundant, Recovery Error and Format.
Alarm - Alarm config	Set motion detection alarm, Blind alarm, alarm in, alarm out, abnormity alarm, Insight, Human and scene change

	System config - System config - System config	Set the hard disk full state, device name, automatic logout time, audio In
	System config- System config - General config	Set the time display mode on the preview screen
	System config- System config - Time Setting	Set NTP timing, Time Zone, Manual Time.
	System config- Security- Security	Set ONVIF Auth on/off and RTSP Auth
	System config - System Tool - User	Add, modify, delete users, modify the device password in the modify user, add, modify, delete group
	System config - System Tool - Auto Maintain	Set automatic restart time and automatically deleted files
	System config - System Tool – System Tools	Restore the device to factory settings and default settings. Reboot Import and export of parameters of the configuration file. Upgrade the firmware of the device
	System Info - System Info	View the version information, web version, extension information, release date, MAC address, and serial number of the device.
	System Info - Log Info	View device operation information

Alarm

Alarm type- Motion Detection, Video Tampering, Loss alarm, Disk error, Disk full, Analysis Alarm, Alarm Input, Human, Camera Shift Operation-Prompt, Alarm Sound Enable-Sound Path.



Alarm type

☐ Motion Detection ☐ Video Tampering

☐ Loss alarm ☐ Disk error

☐ Disk full ☐ Analysis alarm

☐ Alarm Input ☐ Human

☐ Camera Shift

Operation

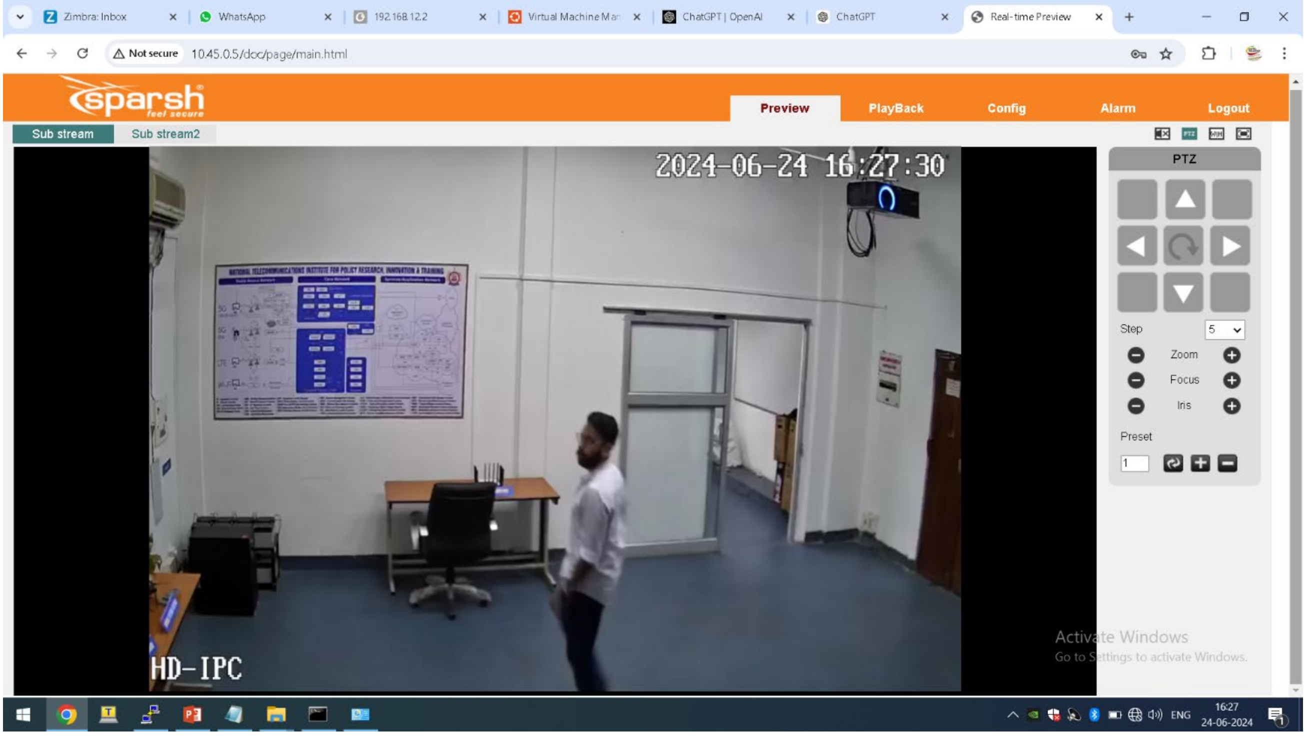
☐ Prompt

Alarm sound

☐ Enable

Sound path [ ] select

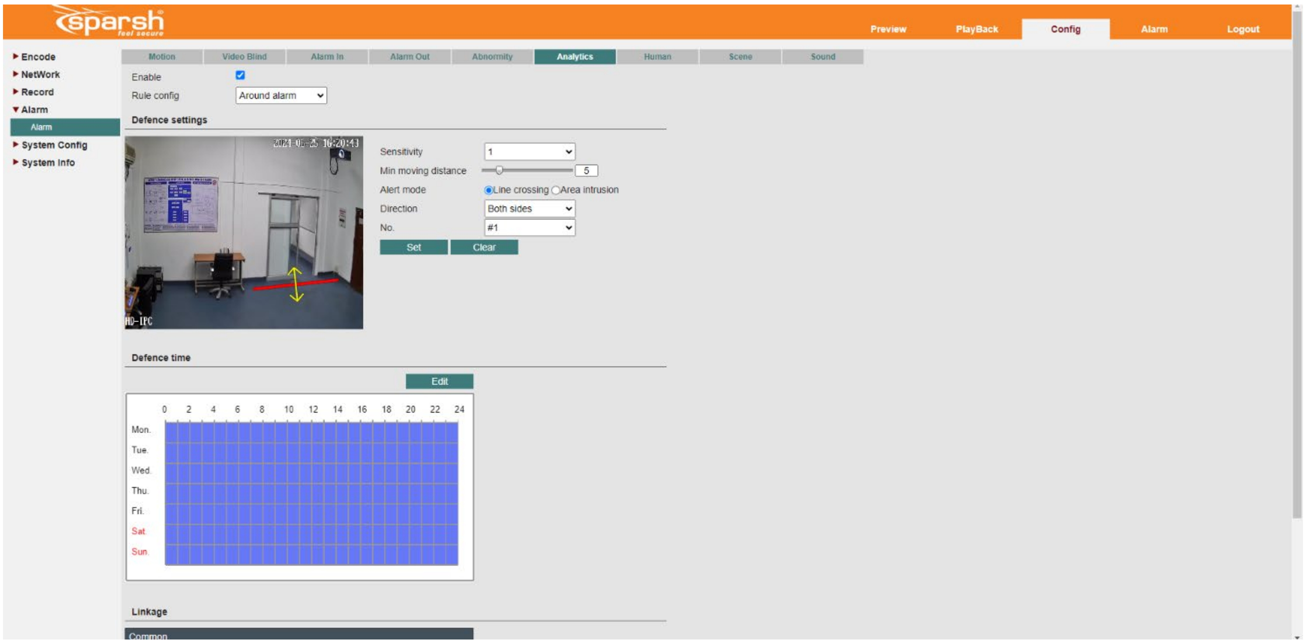
No.	Type	Time	Channel



Step 5

Preset 1

Live Alarm in case of crossing



Enable ☑

Rule config: Around alarm

Defence settings

Sensitivity: 1

Min moving distance: 5

Alert mode: ● Line crossing ○ Area intrusion

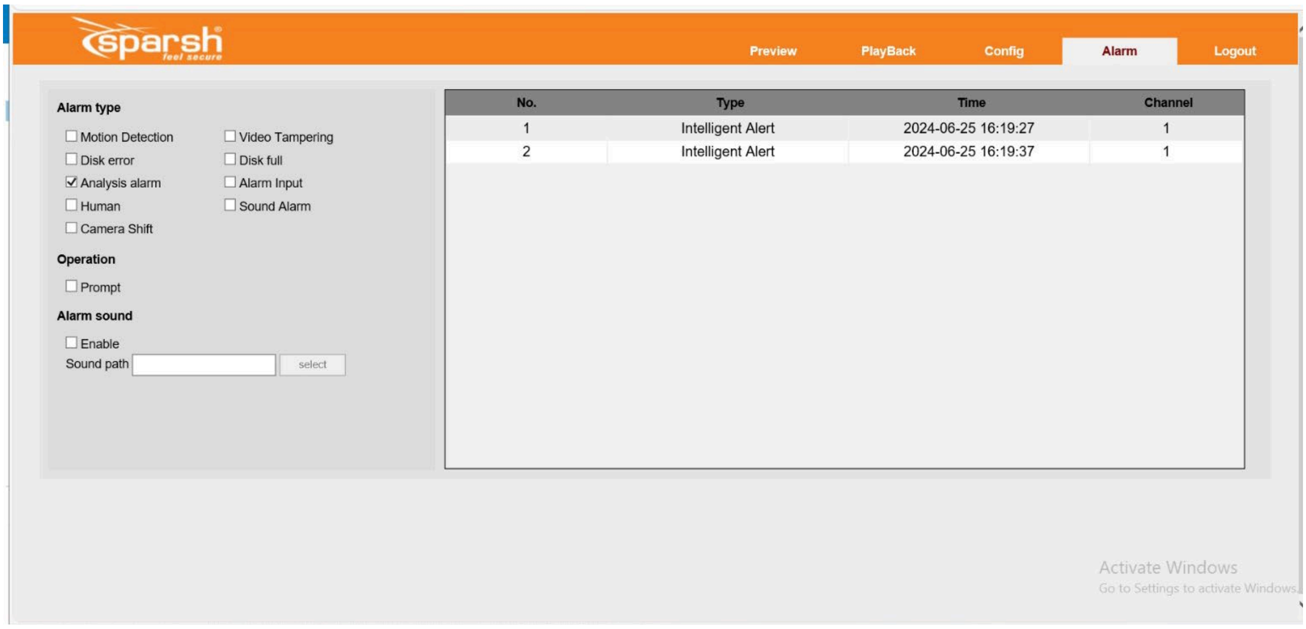
Direction: Both sides

No.: #1

Defence time

	0	2	4	6	8	10	12	14	16	18	20	22	24
Mon.													
Tue.													
Wed.													
Thu.													
Fri.													
Sat.													
Sun.													

Check the alarm logs by checking the analysis alarm section in the upper tab section



Alarm type

☐ Motion Detection ☐ Video Tampering

☐ Disk error ☐ Disk full

☑ Analysis alarm ☐ Alarm Input

☐ Human ☐ Sound Alarm

☐ Camera Shift

Operation

☐ Prompt

Alarm sound

☐ Enable

Sound path

No.	Type	Time	Channel
1	Intelligent Alert	2024-06-25 16:19:27	1
2	Intelligent Alert	2024-06-25 16:19:37	1

Real-Time Fire Detection Using a 5G AI Camera

Overview

AI-powered fire detection using a 5G-enabled camera & MEC-based VM.
Uses YOLOv5/YOLOv9 for real-time inference.
Ensures low latency, instant alerts, and remote monitoring.

Implementation Steps

Set Up VM on MEC Server

OS: Ubuntu 20.04/22.04 LTS
Resources: 10+ cores CPU, 16GB RAM, 100GB SSD
Install dependencies:

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y git python3 python3-pip ffmpeg
pip3 install torch torchvision torchaudio opencv-python numpy
```

Configure 5G Camera Connectivity

Assign static IP to VM:

IP: 192.168.xx.xx
Subnet: 255.255.255.0
Gateway: 192.168.x.10 (5G Core)

Deploy Fire Detection Model

Find & Clone YOLO-based Fire Detection Repository

```
git clone <repository_url>

cd <repository_name>

pip3 install -r requirements.txt
```

- Process RTSP video stream for fire detection in real-time.

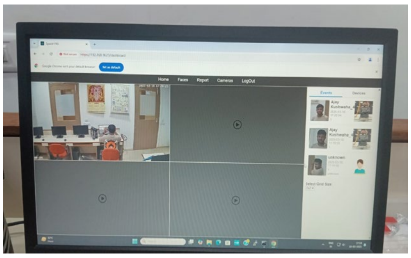
Benefits

-Real-time alerts for faster response.
-Low-latency processing with MEC.
-Remote monitoring & scalability.

Challenges & Future Enhancements

- Network dependency: Requires stable 5G.
- False positives: Needs model fine-tuning.
- Optimizations: Edge AI for faster inference & improved accuracy.

Restricted Access Control System Using a 5G AI Camera



Overview

1. AI-powered facial recognition for secure access control.
2. Uses a 5G-enabled camera and MEC-based processing for real-time recognition.
3. Enhances security, minimizes unauthorized entry, and automates access logs.

System Components

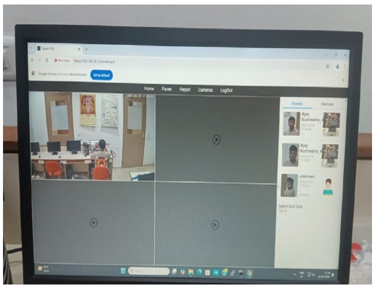
1. 5G-Enabled Camera – Captures real-time video feed.
2. MEC Server – Runs AI facial recognition for access validation.
3. Access Control Mechanism – Electronic door lock integrated with the system.
4. Cloud Database (Optional) – Stores personnel data and access logs.

Key Features

1. Real-time facial recognition ensures only authorized entry.
2. Automated alerts notify in case of unauthorized access attempts.
3. Scalable deployment across multiple locations with centralized monitoring.

Implementation Process

1. Camera & Network Setup – Configure 5G camera and RTSP streaming.
2. MEC Server Installation – Install AI-driven facial recognition software.
3. Database Configuration – Store personnel images and access levels.
4. Access Control Integration – Link AI system to electronic locks.
5. Real-time Monitoring – Log each access attempt for security.



Challenges & Solutions

1. Network Latency Issues – Ensure stable 5G connectivity.
2. False Positives/Negatives – Optimize AI with better datasets.
3. RTSP Streaming Issues – Verify network and camera firmware.

Future Enhancements

1. Edge AI integration for even faster recognition.
2. Improved AI training for better accuracy.
3. Cloud-based analytics for enhanced monitoring.

5G Mini Drone

1. Overview

Suparna control stack runs on autopilot hardware to control drones, UAVs, and other unmanned vehicles. It offers robust capabilities for controlling a wide range of vehicles like multi-copters, fixed-wing, VTOLs, and ground vehicles. Here a quadcopter with autonomous capabilities is presented

2. System Components

The system is composed of several key components that work together to ensure smooth flight operations and management of the vehicle. These components include the Flight Stack, Middleware, Hardware Abstraction, and the Operating System.

3. Flight Stack

The Flight Stack includes the navigation, position estimation, and attitude controllers. It is responsible for ensuring the vehicle follows flight paths, maintains stability, and reaches its intended destination.

4. Middleware

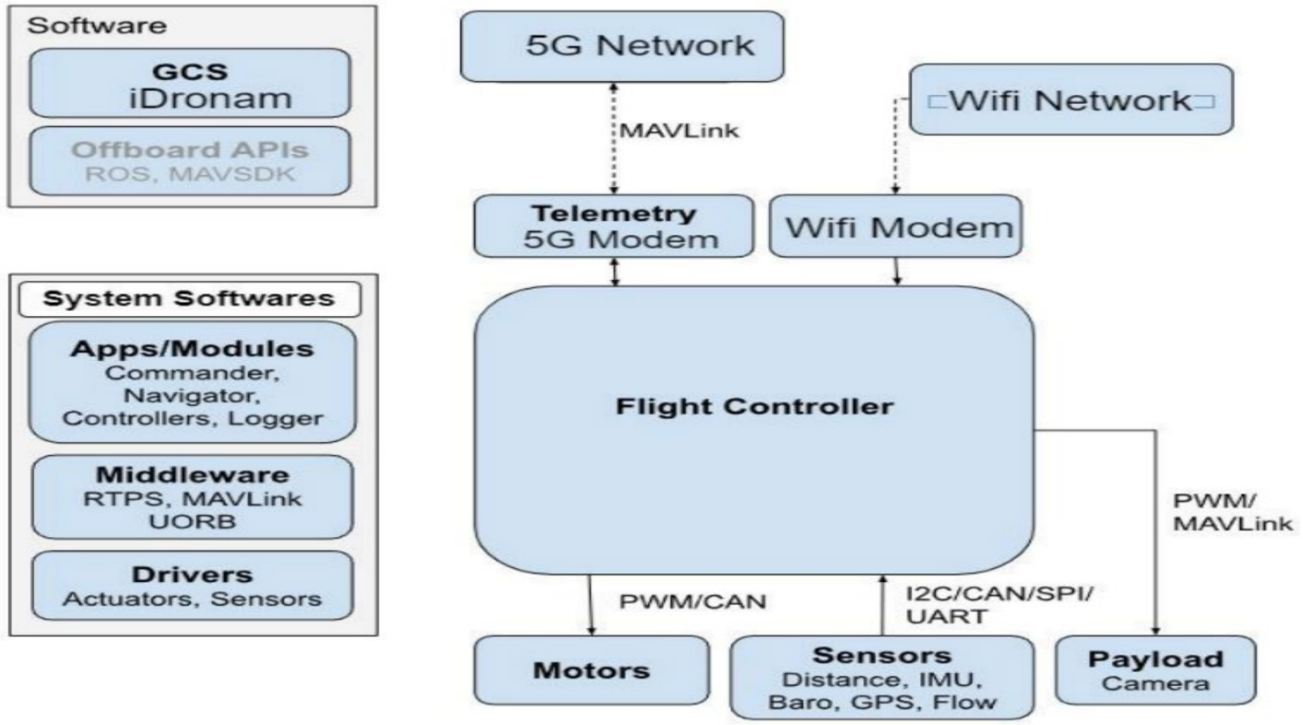
PX4's Middleware facilitates communication between different parts of the system, such as the flight control and sensors. It provides standard interfaces for vehicle components, making the system more modular and extensible.

5. Hardware Abstraction Layer (HAL)

The Hardware Abstraction Layer separates the operating system and hardware-specific implementations from the higher-level flight logic. This ensures that PX4 can run on various autopilot hardware with minimal changes to the software.

6. Operating System

The controller is designed to run on top of a real-time operating system, providing low-latency, predictable scheduling required for flight control applications.



Mission Protocol

The mission sub-protocol allows a GCS or developer API to exchange mission (flight plan), geofence and safe point information with a drone/component.

The protocol covers:

Operations to upload, download and clear missions, set/get the current mission item number, and get notification when the current mission item has changed.

Message type(s) and enumerations for exchanging mission items.

Mission Items ("MAVLink commands") that are common to most systems.

The protocol supports re-request of messages that have not arrived, which allows missions to be reliably transferred over a lossy link.

CONDITION commands (MAV_CMD_CONDITION_*) for changing the execution of the mission based on a condition - e.g. pausing the mission for a time before executing next command (MAV_CMD_CONDITION_DELAY).

Geofence mission items:

Prefixed with MAV_CMD_NAV_FENCE_ (e.g. MAV_CMD_NAV_FENCE_RETURN_POINT).

Rally point mission items:

There is just one rally point MAV_CMD: MAV_CMD_NAV_RALLY_POINT.

The commands are transmitted/encoded in MISSION_ITEM or MISSION_ITEM_INT messages. These messages include fields to identify the particular mission item (command id) and up to 7 command-specific optional parameters.

Field Name	Type	Values	Description
command	uint16_t	MAV_CMD	Command id, as defined in MAV_CMD.
param1	float		Param #1.
param2	float		Param #2.
param3	float		Param #3.
param4	float		Param #4.
param5 (x)	float / int32_t		X coordinate (local frame) or latitude (global frame) for navigation commands (otherwise Param #5).
param6 (y)	float / int32_t		Y coordinate (local frame) or longitude (global frame) for navigation commands (otherwise Param #6).
param7 (z)	float		Z coordinate (local frame) or altitude (global - relative or absolute, depending on frame) (otherwise Param #7).

The first four parameters (shown above) can be used for any purpose - this depends on the particular command. The last three parameters (x, y, z) are used for positional information in MAV_CMD_NAV_* commands, but can be used for any purpose in other commands.

The remaining message fields are used for addressing, defining the mission type, specifying the reference frame used for x, y, z in MAV_CMD_NAV_* messages, etc.:

Field Name	Type	Values	Description
target_system	uint8_t		System ID
target_component	uint8_t		Component ID

seq	uint16_t		Sequence number for item within mission (indexed from 0).
frame	uint8_t	MAV_FRAME	The coordinate system of the waypoint.

PX4 support global frames in MAVLink commands (local frames may be supported if the same command is sent via the command protocol).

mission_type	uint8_t	MAV_MISSION_TYPE	Mission type.
current	uint8_t	false:0, true:1	When downloading, whether the item is the current mission item.
autocontinue	uint8_t		Autocontinue to next waypoint when the command completes.

MISSION_ITEM_INT vs MISSION_ITEM

MISSION_ITEM and MISSION_ITEM_INT are used to exchange individual mission items between systems. MISSION_ITEM messages encode all mission item parameters into float parameters fields (single precision IEEE754) for transmission. MISSION_ITEM_INT is exactly the same except that param5 and param6 are Int32 fields.

Protocol implementations must allow both message types in supported operations (along with the corresponding MISSION_REQUEST and MISSION_REQUEST_INT message types).

For detailed information, it is advisable to refer the manual of SUPARNA Drone provided for the equipment.

Real-Time Fire Detection Using a 5G Drone

Overview

- Uses a 5G-enabled drone and an MEC (Multi-access Edge Computing) server for real-time fire detection.
- AI-powered analysis using YOLOv5/v9 ensures low-latency, high-accuracy detection.
- Ideal for industrial, commercial, and remote area surveillance.

Step 1: Set Up a Virtual Machine (VM) on the MEC Server

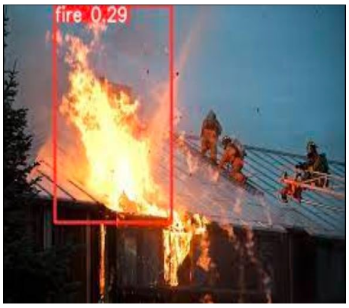
- OS: Ubuntu 20.04/22.04 LTS (stable for AI & deep learning).
- Resources:
  - CPU: 10 cores (8+ recommended for real-time processing).
  - RAM: 12GB (16GB+ recommended).
  - Storage: 100GB SSD.
- Install Dependencies for deep learning and video processing.

Step 2: Establish Network Connectivity with the 5G Drone

- The drone streams video over RTSP (Real-Time Streaming Protocol).
- Configure the VM with a static IP to communicate with the 5G Core Gateway.
  - Example:
    - IP: 192.168.xx.xx
    - Subnet Mask: 255.255.255.0
    - Gateway: 192.168.x.10

Step 3: Implement Fire Detection System

5G Drone Setup



- Equipped with an HD thermal camera.
- Streams real-time surveillance footage.

AI-Powered Fire Detection

- The MEC server processes live video streams using YOLOv5/v9.
- Detects flames and triggers visual alerts in real time.

Conclusion

High-Speed, Low-Latency Surveillance – AI detects fires instantly.
Real-Time Alerts – Immediate response to potential hazards.
Remote Monitoring – Seamless RTSP streaming over 5G.
Wide Applications – Ideal for industrial, commercial, and remote safety monitoring.

Future Optimizations:

- Edge AI integration for faster, on-device processing.
- Enhanced AI model training for improved detection accuracy.

Real-Time Human Detection Using a 5G Drone

Overview

- Uses a 5G-enabled drone and an MEC (Multi-access Edge Computing) server for real-time human detection.
- AI-powered YOLOv5/v9 model ensures low-latency, high- accuracy tracking.
- Ideal for security, surveillance, and access control.

Step 1: Set Up a Virtual Machine (VM) on the MEC Server

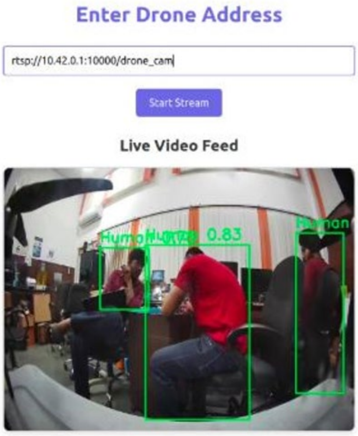
- OS: Ubuntu 20.04/22.04 LTS (stable & AI-compatible).
- Resources:
  - CPU: 10 cores (8+ recommended for real-time AI processing).
  - RAM: 12GB (16GB+ recommended).
  - Storage: 100GB SSD.
- Install Dependencies (AI libraries, deep learning frameworks).
- Optional: Use Docker inside the VM for flexibility.

Step 2: Establish Network Connectivity with the 5G Drone

- The drone streams video over RTSP (Real-Time Streaming Protocol).
- Configure the VM with a static IP for seamless 5G Core Gateway communication.
  - Example:
    - IP: 192.168.xx.xx
    - Subnet Mask: 255.255.255.0
    - Gateway: 192.168.x.10

Step 3: AI-Based Human Detection Model Deployment
Selecting the Right AI Model

- Uses YOLOv5/v9 trained for human detection from aerial views.
- Differentiates between humans and background elements.



Enter Drone Address

rtsp://10.42.0.1:10000/drone_cam

Real-Time Processing on MEC Server

- AI model analyzes RTSP video feed in real time.
- Detects human presence, tracks movements, and flags anomalies.
- Enables security alerts and access control based on detections.

Conclusion

Fast & Accurate Surveillance – AI enhances security and monitoring.

Real-Time Alerts – Detects unauthorized individuals instantly.

5G-Powered Streaming – Ensures smooth and continuous video processing.

Scalable & Secure – Ideal for industrial, commercial, and security applications.

Future Optimizations:

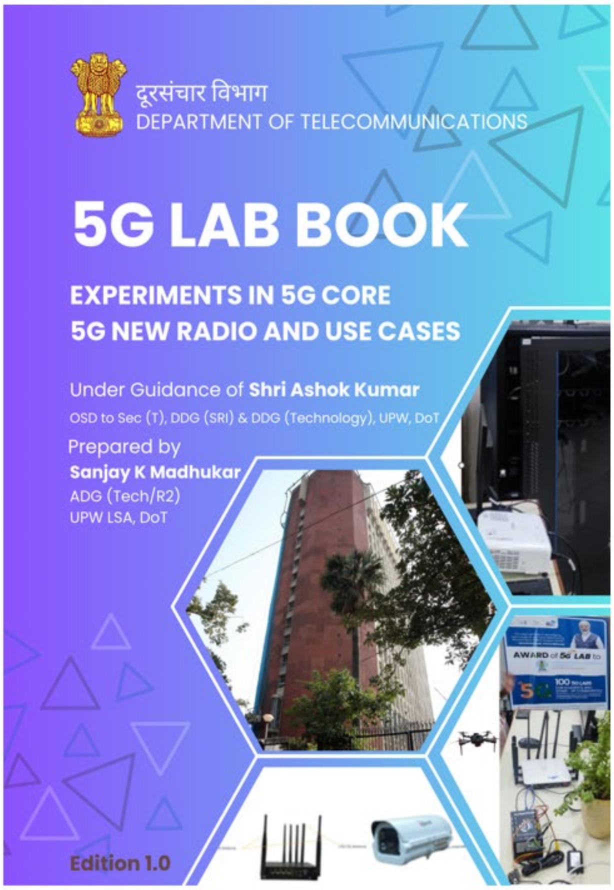
- Enhanced AI training for improved detection accuracy.
- Integration with biometric recognition for access control.

As we close this journey through the architecture and aspirations of 5G, we recognize that every generation of technology is both a culmination and a beginning. The waves we ride today will ripple into tomorrow's networks.

The spectrum may be finite, but our imagination is not. With 5G, we've only just begun to reimagine what's possible.

And so we built one. May this book serve as both a map and a mirror for those shaping the networks of tomorrow as well as bridge—between bytes and Bharat, between policy and people.

With Best Wishes,
- **Sanjay Kr Madhukar**
ADG, Department of Telecommunications



दूरसंचार विभाग
DEPARTMENT OF TELECOMMUNICATIONS

5G LAB BOOK

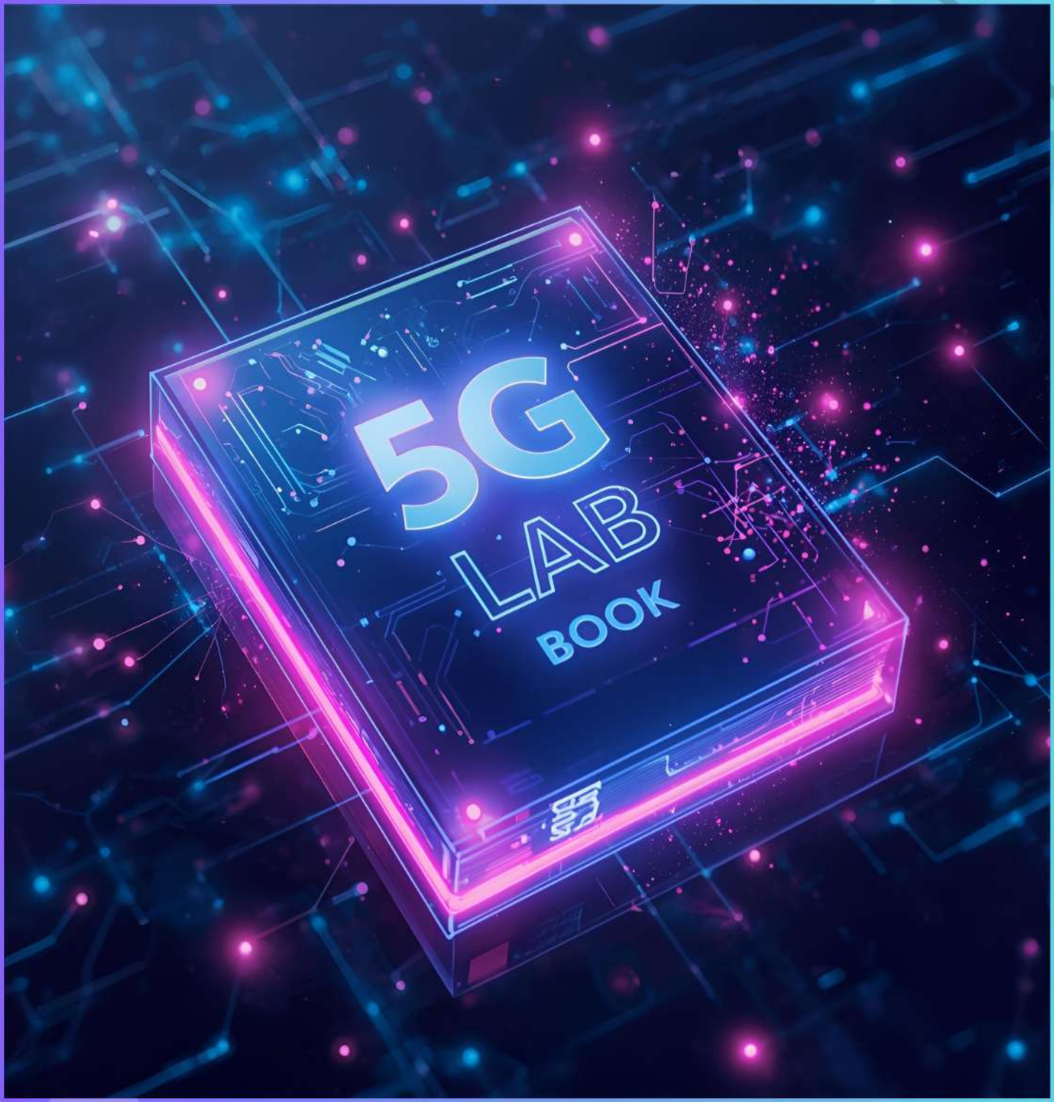
EXPERIMENTS IN 5G CORE
5G NEW RADIO AND USE CASES

Under Guidance of **Shri Ashok Kumar**
OSD to Sec (T), DDG (SRI) & DDG (Technology), UPW, DoT

Prepared by
Sanjay K Madhukar
ADG (Tech/R2)
UPW LSA, DoT

Edition 1.0

दूरसंचार विभाग
DEPARTMENT OF TELECOMMUNICATIONS



UP West LSA, DoT

Edition 1.0