ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA I.C.A.I.

Máster en Ingeniería Industrial - MII

PROYECTO FIN DE MÁSTER

ESTUDIO DE LA SEGURIDAD EN REDES MÓVILES 5G Y VECTORES DE ATAQUE A LA IDENTIDAD DE LOS ABONADOS

Autor - ***Miguel Gallego Vara***
Director - ***Pedro Cabrera Cámara***

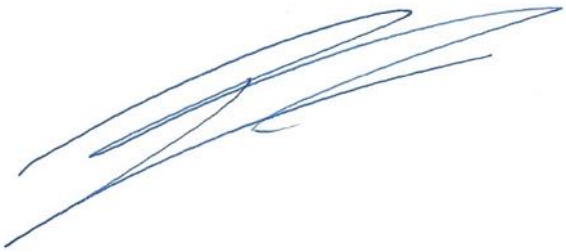
Madrid
Julio de 2021

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título
ESTUDIO DE LA SEGURIDAD EN LAS REDES MÓVILES 5G Y VECTORES DE ATAQUE A LA IDENTIDAD DE LOS ABONADOS
en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico **2020-2021** es de mi autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos. El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

Fdo.:  Fecha: **30/06/2021**

Autorizada la entrega del proyecto
EL DIRECTOR DEL PROYECTO

Fdo.: Pedro Cabrera Cámara Fecha: **30/06/2021**



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA I.C.A.I.

Máster en Ingeniería Industrial - MII

PROYECTO FIN DE MÁSTER

ESTUDIO DE LA SEGURIDAD EN REDES MÓVILES 5G Y VECTORES DE ATAQUE A LA IDENTIDAD DE LOS ABONADOS

Autor - ***Miguel Gallego Vara***
Director - ***Pedro Cabrera Cámara***

Madrid
Julio de 2021

Agradecimientos

Este proyecto no se habría podido realizar sin el constante apoyo de Pedro, director del trabajo y fundador de Ethon Shield, empresa colaboradora con este trabajo, muchas gracias.

ESTUDIO DE LA SEGURIDAD EN REDES MÓVILES 5G Y VECTORES DE ATAQUE A LA IDENTIDAD DE LOS ABONADOS

Autor: Gallego Vara, Miguel.
Director: Cabrera Cámara, Pedro.
Entidad Colaboradora – Ethon Shield SL.

RESUMEN DEL PROYECTO

Introducción

La tecnología 5G promete ser la solución a la gran demanda de servicios no solo por parte de la industria si no también de los usuarios de a pie. Las velocidades que podrán alcanzar los dispositivos conectados a estas redes móviles superarán con creces las proporcionadas por su predecesora, la cuarta generación, permitiendo no solo el acceso a Internet a usuarios con teléfonos móviles si no también a dispositivos IoT, coches o maquinaria de industria, comunicándose en tiempo real. A lo largo de los últimos años, ha existido mucha especulación acerca de la seguridad de esta nueva tecnología, poniéndola en duda. El 5G conectará a miles de dispositivos en una red muy compleja, por lo que la superficie de ataque será mucho mayor. No solo eso, si no que estas redes soportarán infraestructura crítica y un ataque a esta infraestructura podrá tener un impacto mucho mayor en la sociedad que en otras tecnologías. Además, los métodos tradicionales de implantación de una red móvil se están modificando, dando un enfoque de virtualización, que hace referencia a ejecutar en software una función que tradicionalmente se ejecutaba en hardware dedicado. Esto permitirá a las redes no solo conseguir una mayor flexibilidad si no también la posibilidad de escalarlas de forma dinámica. Dos de los conceptos relacionados con la virtualización que han revolucionado muchos aspectos de las redes 5G son SDN y NFV. Por tanto, se estudiará la seguridad de las redes 5G alrededor de los siguientes pilares: privacidad de los usuarios, seguridad de la virtualización y seguridad de los miles de nuevos dispositivos conectados. Por otra parte, este proyecto se ha focalizado en uno de los ataques con más renombre hacia las tecnologías móviles: el IMSI Catcher. La finalidad de este ataque es sustraer la identidad del abonado, o IMSI, para así determinar su localización, además de poder perpetrar otros numerosos ataques a partir de la obtención de este número. Este valor identifica inequívocamente al abonado por lo que su captura es considerada una vulnerabilidad crítica. La idea detrás de este ataque es simular una red móvil o estación base y hacer que el abonado se intente conectar a esta red. No es necesario ni que se conecte puesto que el cliente se identifica con el IMSI en el procedimiento de acople a la red, aunque este abonado no este registrado en la base de datos y posteriormente sea rechazado. Este ataque se comprobará en los dos despliegues principales que tiene la quinta generación.

La estructura de las redes 5G, al igual que sus predecesoras, se dividen en tres componentes principales:

- **UE** – Dispositivo que alberga la tarjeta SIM que identifica de forma única al cliente. En las tecnologías anteriores siempre han correspondido a teléfonos móviles o tablets, pero a partir de ahora podrán ser desde coches hasta maquinaría del sector industrial.
- **NG-RAN** – La parte radio, la arquitectura que se ha modificado considerablemente para aumentar las velocidades de subida y de bajada, con el uso de frecuencias altas. El nodo que se encarga de comunicarse con el UE se denomina gNB.

- **NG-MC** – Es el core de la red donde se combinan múltiples funcionalidades entre las que encontramos: proporcionar comunicación con Internet, permitir la movilidad del usuario y facturación. Todo ello, asegurando seguridad y un QoS.

3GPP plantea múltiples escenarios de despliegue de una red 5G que se engloban en dos categorías principales: NSA y SA. El despliegue NSA permite al 5G apoyarse en la ya existente infraestructura 4G-LTE. Puesto que se apoya en la tecnología anterior, el UE puede conectarse tanto al nodo radio eNB como el gNB. Entre estos dos nodos, uno de ellos actua como el nodo maestro, mientras que el otro se denomina nodo secundario. Por otra parte, el despliegue SA será completamente 5G. Por ello, tanto la infraestructura radio como el core de la red tendrá que ser implementada. Hoy en día es prácticamente imposible encontrar alguna red comercial 5G-SA, puesto que todavía se están acabando de definir parte de los estándares, aunque si es cierto que se han desarrollado algunas redes piloto.

Metodología

Par poder desplegar una red móvil en el laboratorio era indispensable tener un conocimiento básico de las redes móviles. Por tanto, la primera parte del proyecto era obtener un conocimiento de las tecnologías precedentes al 5G, sobre todo del 4G – LTE, puesto que uno de los despliegues se apoya en esta tecnología. La principal referencia para estudiar en detalle tanto 4G como 5G fue la documentación publicada por 3GPP, organización que desarrolla los estándares en las telecomunicaciones, además de ciertas referencias especificadas en la bibliografía. Además, para la realización del proyecto se han utilizado diversas herramientas. En primer lugar, se ha basado en software libre desarrollado por Open Air Interface (OAI), una plataforma de proyectos de redes de acceso radio (RAN) y redes core (CN). Esta organización, que comenzó con el desarrollo de 4G – LTE, durante los últimos años, decidió centrarse en la ejecución de los distintos despliegues de la quinta generación. A julio de 2021, el desarrollo del 5G – NSA está completo, sin embargo, el despliegue 5G-SA todavía está en desarrollo. Por otra parte, se utilizarán varios dispositivos SDR, en concreto, los desarrollados por Ettus Research. Estos dispositivos permiten la configuración mediante software lo que tradicionalmente se ha ejecutado en hardware, obteniendo una mayor flexibilidad en los desarrollos. Para el UE, se compró un Oppo Reno 4z 5G, además de las tarjetas SIM reprogramables proporcionadas por sysmocom. Finalmente, para el análisis de las distintas trazas entre los nodos de la red, se utilizó Wireshark, herramienta muy útil para comprobar el tráfico de las comunicaciones.

Resultados

Como se comprobará en los resultados, debido al desarrollo de OAI, solo se pudo comprobar el IMSI Catcher en 5G – NSA. La experimentación del despliegue apoyado en 4G confirmó la posibilidad de perpetrar un IMSI Catcher, en concreto, en la opción de despliegue dónde el eNB es el nodo maestro y los nodos radio están conectados al core 4G. Esto es debido a que la primera fase del procedimiento de acceso a la red en 5G-NSA es la misma que en la tecnología precedente. Cuando se establece conexión con el nodo radio "*antiguo*", es decir el eNB, es cuando comienza la transferencia al nodo de la siguiente generación, por tanto, el envío de la identidad del abonado sigue los mismos protocolos que en 4G. Desplegando una red de laboratorio 5G-NSA, se podía obtener el IMSI del dispositivo móvil en cuestión examinando los mensajes intercambiados entre el nodo radio y el core de la red. Por otra parte, en las red 5G – SA proporcionada por OAI, el UE solo intercambiaba un par de mensajes con el nodo radio, sin poder registrarse ni enviar su identidad, es decir, no enviaba una solicitud de acople

en ningún momento. No obstante, en 5G-SA, 3GPP especifica varias herramientas por las cuales el IMSI Catcher no se podría realizar. Esto es debido a que la tarjeta SIM 5G utilizan un cifrado asímetrico, donde se utiliza una clave pública y privada de la operadora. La clave pública de la operadora se almacena en la tarjeta SIM, mientras la red guarda la clave privada, cifrando las comunicaciones desde el inicio. Sin embargo, la teoría dista mucho de la práctica y la implementación de esto en las redes móviles públicas puede no ser una realidad. El estudio de las redes 5G-SA queda para el futuro, cuando OAI desarrolle este despliegue. Por otra parte, se ha comprobado como el despliegue del 5G de la operadora Movistar, sobre todo en la Comunidad de Madrid, está limitada práctimante en su totalidad a 5G-NSA. Esto es debido a que la infraestructura 4G de las operadoras principales está prácticamente desplegada por todo el país, por lo que tiene sentido que el avance natural sea desplegar en primera instancia el 5G-NSA.

Conclusiones

Las principales conclusiones obtenidas con este proyecto han sido diversas. En primer lugar, los IMSI Catchers seguirán apareciendo en esta nueva tecnología, sobre todo porque 5G-NSA será dominante en estos primeros años de transición. En 5G-SA todavía será una incógnita puesto que las especificaciones descritas por 3GPP exigen que los usuarios no envíen su identidad en texto claro a través de la red. Por ello, habrá que comprobar en los estudios futuros la implementación de la seguridad propuesta por las entidades principales para prevenir el robo de las identidades de los usuarios. Existirán dos posibilidades, a través de software libre como OAI y srsRAN o con el uso de soluciones comerciales como Amarisoft o YateBTS. Por otra parte, el estado de implementación de las redes 5G está en la fase alpha, es decir, comenzando. Todos los despliegues 5G son NSA puesto que 5G-SA todavía esta en fase de desarrollo; faltan muchas especificaciones por desarrollar. En cuanto a la securización de las redes, las tecnologías móviles hoy en día son mucho más seguras que en su comienzo, y en el 5G no es diferente, siguen existiendo protocolos de seguridad importantes y se han añadido nuevas funcionalidades. Por otra parte, en la teoría la virtualización parece ser la mayor innovación en las redes 5G, permitiendo una gran flexibilidad y escalabilidad, sin embargo, su implementación será una incógnita. Las operadoras tendrán que proporcionar un ecosistema seguro y fiable puesto que aparecerán nuevas posibilidades de ataques a esta infraestructura.

STUDY OF SECURITY IN 5G MOBILE NETWORKS AND ATTACK VECTORS ON SUBSCRIBER IDENTITIES

Introduction

5G technology promises to be the solution to the great demand for services not only from the industry but also from ordinary users. The speeds that devices connected to these mobile networks will be able to reach will far exceed those provided by its predecessor, the fourth generation, allowing not only Internet access to users with cell phones but also to IoT devices, cars or industrial machinery, communicating in real time. Over the last few years, there has been much speculation about the security of this new technology, calling it into question. 5G will connect thousands of devices in a very complex network, so the attack surface will be much larger. Not only that, but these networks will support critical infrastructure and an attack on this infrastructure could have a much greater impact on society than on other technologies. In addition, the traditional methods of deploying a mobile network are changing, with a focus on virtualization, which refers to running in software a function that traditionally ran on dedicated hardware. This will allow networks not only to achieve greater flexibility but also the ability to scale dynamically. Two of the concepts related to virtualization that have revolutionized many aspects of 5G networks are SDN and NFV. Therefore, the security of 5G networks will be studied around the following pillars: user privacy, virtualization security and security of the thousands of new connected devices. On the other hand, this project has focused on one of the most renowned attacks on mobile technologies: the IMSI Catcher. The purpose of this attack is to steal the subscriber's identity, or IMSI, in order to determine its location, as well as to perpetrate numerous other attacks based on obtaining this number. This value unequivocally identifies the subscriber, so obtaining it is considered a critical vulnerability. The idea behind this attack is to simulate a mobile network or base station and make the subscriber try to connect to this network. It is not even necessary to connect since the client identifies itself with the IMSI in the network attachment procedure, even though this subscriber is not registered in the database and is subsequently rejected. This attack will be tested in the two main deployments of the fifth generation.

The structure of 5G networks, like their predecessors, are divided into three main components:

- **UE** - Device that houses the SIM card that uniquely identifies the customer. In previous technologies they have always corresponded to cell phones or tablets, but from now on they may range from cars to machinery in the industrial sector.
- **NG-RAN** - The radio part, the architecture that has been considerably modified to increase upload and download speeds, with the use of high frequencies. The node that communicates with the UE is called the gNB.
- **NG-MC** - This is the core of the network where multiple functionalities are combined, including: providing communication with the Internet, enabling user mobility and billing. All this while ensuring security and QoS.

3GPP proposes multiple 5G network deployment scenarios that fall into two main categories: NSA and SA. The NSA deployment allows 5G to build on the existing 4G-LTE infrastructure. Since it relies on the previous technology, the UE can connect to both the eNB and the gNB radio node. Between these two nodes, one of them acts as the master node, while the other is referred to as the secondary node. On the other hand, the SA deployment will be fully 5G. Therefore, both the radio infrastructure and the network core will have to be implemented. Today it is practically impossible to find any commercial 5G-SA network, since part of the

standards are still being defined, although it is true that some pilot networks have been developed.

Methodology

In order to deploy a mobile network in the laboratory, it was essential to have a basic knowledge of mobile networks. Therefore, the first part of the project was to obtain a knowledge of the technologies preceding 5G, especially 4G - LTE, since one of the deployments is based on this technology. The main reference to study in detail both 4G and 5G was the documentation published by 3GPP, the organization that develops standards in telecommunications, in addition to certain references specified in the bibliography. In addition, several tools were used to carry out the project. Firstly, it was based on free software developed by Open Air Interface (OAI), a platform for radio access network (RAN) and core network (CN) projects. This organization, which started with the development of 4G - LTE, during the last few years, decided to focus on the implementation of the various deployments of the fifth generation. As of July 2021, the 5G - NSA development is complete, however, the 5G-SA deployment is still under development. Moreover, several SDR devices will be used, namely those developed by Ettus Research. These devices allow configuration via software what has traditionally been executed in hardware, obtaining greater flexibility in developments. For the UE, an Oppo Reno 4z 5G was purchased, in addition to the reprogrammable SIM cards provided by sysmocom. Finally, for the analysis of the different traces between the network nodes, Wireshark was used, a very useful tool to check the communications traffic.

Results

As will be seen in the results, due to the development of OAI, only the IMSI Catcher could be tested in 5G - NSA. The experimentation of the 4G supported deployment confirmed the possibility of perpetrating an IMSI Catcher, concretely in the deployment option where the eNB node corresponds to the master node and both nodes are connected to the 4G core. This is because the first phase of the network access procedure in 5G-NSA is the same as in the preceding technology. When connection is established with the "old" radio node, i.e. the eNB, is when the handover to the next generation node begins, therefore, the sending of the subscriber's identity follows the same protocols as in 4G. By deploying a 5G-NSA laboratory network, the IMSI of the mobile device in question could be obtained by examining the messages exchanged between the radio node and the network core. On the other hand, in the 5G - SA network provided by OAI, the UE only exchanged a couple of messages with the radio node, without being able to register or send its identity, i.e., it did not send a coupling request at any time. However, in 5G-SA, 3GPP specifies several tools by which the IMSI Catcher could not be performed. This is because the 5G SIM card uses an asymmetric encryption, where an operator's public and private key is used. The operator's public key is stored on the SIM card, while the network stores the private key, encrypting communications from the start. However, theory is far from practice and the implementation of this in public mobile networks may not be a reality. The study of 5G-SA networks remains for the future, when OAI will develop this deployment. On the other hand, it has been verified how the 5G deployment of the operator Movistar, especially in the Community of Madrid, is almost entirely limited to 5G-NSA. This is because the 4G infrastructure of the main operators is practically deployed throughout the country, so it makes sense that the natural progression is to deploy 5G-NSA in the first instance.

Conclusions

The main conclusions obtained with this project have been diverse. First of all, IMSI Catchers will continue to appear in this new technology, especially since 5G-NSA will be dominant in these first years of transition. In 5G-SA it will still be an unknown since the specifications described by 3GPP require that users do not send their identity in clear text over the network. Therefore, it will be necessary to check in future studies the implementation of the security proposed by the main entities to prevent the theft of users' identities. There will be two possibilities, through free software such as OAI and srsRAN or with the use of commercial solutions such as Amarisoft or YateBTS. On the other hand, the state of implementation of 5G networks is in the alpha phase, i.e. just starting. All 5G deployments are NSA since 5G-SA is still in the development phase; many specifications are still to be developed. As for network securitization, mobile technologies today are much more secure than at their inception, and in 5G it is no different, important security protocols still exist and new functionalities have been added. On the other hand, in theory virtualization seems to be the biggest innovation in 5G networks, allowing great flexibility and scalability, however, its implementation will be an unknown. Operators will have to provide a secure and reliable ecosystem as new possibilities for attacks on this infrastructure will emerge.

Índice

Lista de Figuras

Lista de Tablas

Lista de Acrónimos

AMF	Core Access and Mobile Management Function
AUSF	Authentication Server Function
BTS	Base Transceiver Station
BSC	Base Station Controller
DHCP	Dynamic Host Configuration Protocol
DoS	Denial of Service
EIR	Equipment Identity Register
eNB	Evolved Node B
EEA	EPA Encryption Algorithm
EUTRAN	Evolved UMTS Terrestrial Radio Access Network
FDD	Frequency Division Duplexing
GEA	GSM GPRS Encryption Algorithm
gNB	Next Generation Node B
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Services
GSM	Global System for Mobile Communications
GUTI	Global Unique Temporary Identifier
HLR	Home Location Register
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
ICCID	Integrated Circuit Card ID
IMEI	International Mobile Equipment Identity
IMEISV	International Mobile Equipment Identity Software Version

IMSI International Mobile Subscriber Identity

ISIM IP Multimedia Services Identity Module

LTE Long Term Evolution

NEA New Radio Encryption Algorithm

NEF Network Exposure Function

NRF NF Repository Function

NSA Non-StandAlone

NSSF Network Slicing Selector Function

MC Mobile Core

MCC Mobile Country Code

MNC Mobile Network Code

MME Mobility Management Entity

MS Mobile Station

MSC Mobile Switching Center

MSIN Mobile Subscription Identification Number

MSISDN Mobile Station Integrated Services Digital Network

OAI Open Air Interface

PCF Policy Control Function

PEI Permanent Equipment Identifier

PGW Packet Data Network Gateway

RAN Radio Access Network

RAND Random

SA StandAlone

SDAP Service Data Adaptation Protocol

SDR Software Defined Radio

SDSF Structured Data Storage Network Function

SGSN Serving GPRS Support Node

SGW Service Gateway

SIM Subscriber Identity Module

SMF Session Management Function

SRES Signed Response

SUCI Subscription Concealed Identifier

SUPI Subscription Permanent Identifier

TDD Time Division Duplexing

UDM Unified Data Management

UDSF Unstructured Data Storage Network Function

UE User Equipment

UEA UMTS Encryption Algorithm

UICC Universal Integrated Circuit Card

UMTS Universal Mobile Terrestrial System

UPF User Plane Function

USRP Universal Software Radio Peripheral

USIM Universal Subscriber Identity Module

3GPP 3rd Generation Partnership Project

1. Introducción

1.1. Colaboración

Este proyecto se va a realizar en colaboración con *Ethon Shield*, empresa de ciberseguridad dedicada a la consultoría informática y centrada en las telecomunicaciones, drones y radio definida por software. *Ethon Shield* realiza auditorías de seguridad y uno de sus pilares son las redes móviles. Por ello, con esta investigación se observará y analizará desde una perspectiva técnica todo lo relacionado con la seguridad del 5G para que en un futuro se puedan auditar este tipo de redes.

1.2. Metodología

La metodología utilizada en este TFM fue en primer lugar familiarizarse con la arquitectura general de las distintas tecnologías móviles, desde la segunda generación (2G) hasta la actual quinta generación (5G). Así se podría obtener una visión global del funcionamiento general de las redes móviles, puesto que su mayoría comparten los mismos principios. Más adelante se indagó en la estructura del 5G, tanto en el despliegue NSA como SA, estudiando los distintos protocolos que se utilizan, interfaces entre los nodos y la nueva parte radio. Además, se investigó el procedimiento de acceso de la red de los abonados puesto que en esta fase es dónde se podría atacar al cliente y perpetrar un IMSI Catcher. La principal referencia para estudiar en detalle tanto 4G como 5G fue la documentación publicada por 3GPP, organización que desarrolla los estándares en las telecomunicaciones. Por otra parte, para este proyecto era fundamental conocer el funcionamiento de los SDR puesto que se estaría en constante contacto con estos dispositivos, en concreto los Ettus USRP B200/B200 mini. Asimismo, se realizo un estudio del software open source de OAI, tanto de la documentación publicada, como el hardware necesario para su posterior instalación.

1.3. Estructura del trabajo

El **Capítulo 2** presenta al lector el marco teórico del TFM con una pequeña introducción de la evolución de la telefonía celular y un posterior énfasis en el 5G. Se detallará la arquitectura propuesta de esta nueva tecnología, con una explicación de las entidades implicadas, la nueva parte radio, los distintos despliegues y la revolución que ha permitido la virtualización. Además, se hará un breve análisis de la seguridad general de las redes móviles, y las nuevas propuestas de mejora en el 5G. Por último, se explicarán en detalle los procedimientos de acople a la red de los dos despliegues principales en 5G, NSA y SA.

El **Capítulo 3** introducirá al lector todo lo relacionado con el estado actual – a 2021 – del 5G, tanto el estado de implantación como el licenciamiento de frecuencias. Además, se especificarán las distintas soluciones tanto comerciales como open source para la tecnología 5G.

El **Capitulo 4** hace un especial énfasis en las distintas herramientas utilizadas para la realización del proyecto. OAI como Open Source pare recrear una red 5G en el laboratorio, SDR para implementar dicha red, Sysmocom como proveedor de las tarjetas SIM y Wireshark para el análisis de trazas entre los distintos nodos de la red.

El **Capitulo 5** se centrará en los posibles ataques a la identidad de los abonados. Se explicará la historia de los IMSI Catcher y la solución propuesta para las redes 5G.

El **Capítulo 6** y el **Capítulo 7** se centrará en el análisis de los resultados obtenidos y unas conclusiones acerca del proyecto. En el **Capítulo 8** se especifica la bibliografía utilizada a lo largo del proyecto.

El **Capítulo 9** estará compuesto del código y configuraciones utilizadas en la solución propuesta para el IMSI Catcher, para que si el lector está interesado en realizar experimentación pueda seguir los pasos de instalación.

2. Marco teórico

2.1. Historia de las redes móviles

Este capítulo va a comenzar con una breve definición de redes propuesta por José Pico García y David Pérez Conde en uno de sus libros: «Una red es un conjunto de entidades funcionales que tiene como objetivo prestar servicios de comunicaciones móviles a sus suscriptores, con los que se comunica mediante una interfaz de radio» [1]. Dichas redes están compuestas por infinidad de nodos, cada uno con su función o funciones particulares e interaccionan entre ellos, no solo transmitiendo datos sino también desde un plano de control. En las redes móviles, el objetivo final es poder permitir a un usuario comunicarse tanto por voz con otro cliente como navegar por Internet, manteniendo siempre una calidad de servicio óptima.

La aparición de las redes móviles supuso un cambio de paradigma en las comunicaciones. A lo largo de los últimos años, éstas han evolucionado a una velocidad inimaginable. Todo comenzó con la primera generación (1G), en la década de los 80, donde la red estaba diseñada para transportar la voz humana de una forma completamente analógica.

En la década de los 90 apareció la segunda generación de redes móviles (2G), que se conocía como GSM. GSM siguió con el diseño analógico, pero optimizaron muchos procesos mediante protocolos y permitieron los SMS. La arquitectura propuesta en la tecnología 2G sirvió como base a las demás tecnologías. Esta red evolucionó a GPRS soportando servicios tanto de voz como de datos – añadiendo la parte de conmutación de paquetes. Esto fue una revolución, puesto que se pudo comenzar a navegar por Internet, aunque de una forma muy restrictiva. Se diferencian dos conceptos importantes, la conmutación de circuitos y la conmutación de paquetes. Cuando se envía información desde un origen a un destino, en la conmutación de circuitos se reserva ese canal de información y se establece una conexión entre los distintos nodos. Sin embargo, en la conmutación de paquetes, puesto que la información se divide en distintos fragmentos, que en principio pueden ir por rutas distintas, no es necesario reservar capacidades. Cada tipo de conmutación tiene sus ventajas, pero como se podrá ver más adelante, a partir de la cuarta generación solo existiría el dominio de paquetes.

Las redes no pararon de evolucionar con el desarrollo de la tercera generación 3G – UMTS dónde seguía existiendo la división entre voz y datos, pero cada vez se daba más énfasis a la capacidad de estos datos y en conseguir una mayor velocidad de navegación por internet. A lo largo de los años fueron mejorando y desarrollando esta tecnología añadiendo más nodos y capacidad, consiguiendo que los usuarios pudiesen navegar por Internet de forma rápida y cómoda.

La cuarta generación de redes móviles no tardó mucho en llegar – o más conocida como 4G LTE, en la que se decidió optar por una red principal enfocada a los datos, mejorando considerablemente la calidad del servicio, así como su capacidad. Por tanto, las llamadas telefónicas se realizaban en el dominio de paquetes, conocido en telecomunicaciones como Voz sobre IP (VoIP). Así, la red móvil se “simplifico”

y se convirtió en completamente IP. Puesto que uno de los despliegues del 5G se apoya en la cuarta generación, es importante realizar un énfasis en esta tecnología. Podemos diferencias tres componentes principales en la red móvil: UE, EUTRAN y EPC.

- **UE** – Es el dispositivo que alberga la tarjeta SIM, denominada en esta tecnología USIM, donde se almacena la información de la identidad del abonado.
- **EUTRAN** – Es la arquitectura encargada de comunicarse via radio con el UE. El nodo destinado a esta función se denominó eNB. Existen interfaces que comunican el eNB con el UE, con otros eNB y con el EPC como se puede comprobar en la Figura 1. Además, se puede comprobar como existen comunicación tanto en el plano de datos ("Traffic") como el plano de control ("LTE signalling).
- **EPC** – Es el core de la red, ilustrado en la Figura 2, compuesto por los siguientes nodos principales:
    - **P-GW** – Es el nodo conectado a Internet
    - **S-GW** – Es el nodo encargado de la facturación además de actuar como un router entre el eNB y el PGW
    - **MME** – Es el cerebro de la red, encargado de hablar con el HSS, de seguridad y movilidad.
    - **HSS** – Es el servidor donde se almacena la información de los usuarios e información para la autenticación

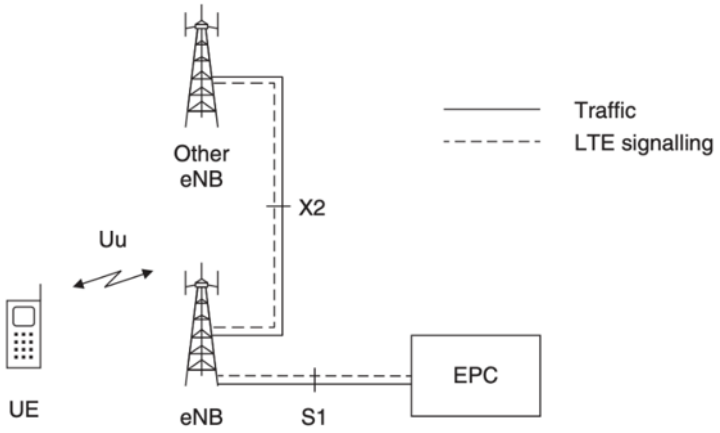


Figura 1: EUTRAN

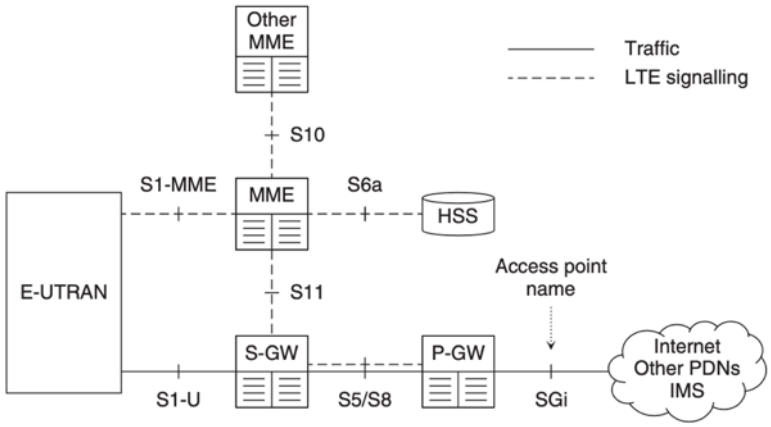


Figura 2: EPC - 4G

En último lugar se desarrolló la quinta generación (5G) de redes móviles detallada en las **Sección 2.2.**

Aunque el despliegue a nivel global del 5G todavía es mínimo ya se empiezan a escuchar las primeras voces acerca de la sexta generación – 6G. Según apuntan expertos, la evolución del 6G se centrará en mejorar todavía más las velocidades de transmisión de los usuarios con el uso de frecuencias muy elevadas y la inteligencia artificial tomará un papel clave en el desarrollo de la infraestructura. Sin embargo, la investigación es a nivel teórico puesto que queda todavía mucho trabajo por desarrollar.

2.2. Quinta generación – 5G

La Quinta generación de redes móviles plantea ser una revolución en las telecomunicaciones permitiendo a todo tipo de equipos estar conectados a Internet o a la nube, pudiendo hablar entre ellos a tiempo real sin apenas latencia. Los estándares aseguran que las velocidades de navegación serán tan elevadas que se podrá descargar una película en cuestión de segundos. Sin embargo, la gran revolución no se ha llevado a cabo exclusivamente en el uso de frecuencias altas para incrementar las velocidades de transmisión de datos, también una gran innovación proviene de un término cada vez más conocido en el mundo de las telecomunicaciones: la virtualización. SDN y NFV – ambas explicadas en detalle en la **Sección 2.3.** – son arquitecturas de virtualización que alejan el hardware de las redes sustituyéndola por software, favoreciendo una mayor flexibilidad y escalabilidad de estas, pudiendo especificar la calidad de servicio de los usuarios de forma selectiva. Esto permitirá manejar de forma dinámica los flujos de datos en horas valle y horas pico, similar a lo que se realiza en el sector eléctrico, reduciendo considerablemente la posibilidad de saturación de la red además de modificar estos flujos dependiendo de la movilidad geográfica de los abonados. Así, en verano, donde de forma general la población en España se mueve a la costa, se podrán trasladar las capacidades a estas zonas geográficas.

Los objetivos del 5G son múltiples, entre ellos encontramos:

- Permitir el avance del Internet de las cosas (IoT – Internet of Things)
- Permitir una disponibilidad mucho mayor, latencia muy baja y movilidad muy alta
- Mejorar considerablemente las velocidades de transmisión de datos
- Consolidar nuevos clientes de las redes industriales

A continuación, se explicará en detalle la arquitectura de las redes 5G, los tipos de despliegue y uno de los aspectos disruptivos de esta tecnología: Network Slicing.

2.2.1. Arquitectura

La estructura de las redes 5G, al igual que sus predecesoras, se dividen en tres componentes principales:

- **UE** – Similar al UE de 4G-LTE. En las tecnologías anteriores siempre han correspondido a teléfonos móviles o tablets, pero a partir de ahora podrán ser desde coches hasta maquinaría del sector industrial.
- **NG-RAN** – La parte radio, la arquitectura que se ha modificado considerablemente para aumentar las velocidades de subida y de bajada, con el uso de frecuencias altas. El nodo que se encarga de comunicarse con el UE se denomina gNB, similar al eNB de 4G-LTE.
- **NG-MC** – Es el core de la red donde se combinan múltiples funcionalidades entre las que encontramos: proporcionar comunicación con Internet, permitir la movilidad del usuario y facturación. Todo ello, asegurando seguridad y un QoS.

2.2.1.1. UE

Los dispositivos evolucionan y los teléfonos móviles no van a ser los únicos conectados a las redes móviles. Hoy en día, en contribución al Internet de las Cosas, los coches, los drones, maquinaria del sector industrial, del sector de la agricultura o incluso electrodomésticos serán capaces de conectarse a Internet mediante la tecnología 5G. Serán capaces de comunicarse entre ellos en tiempo real, o descargar información de Internet a velocidades muy altas. Para conectarse a una red móvil, como se ha mencionado anteriormente, es necesaria una tarjeta SIM. En ella es donde reside toda la información específica para la autenticación e identificación del usuario.

El UE está formado por dos entidades diferentes:

- **ME** – Dispositivo hardware donde reside la UICC y desde donde se realizan/reciben llamadas y se puede navegar por Internet. Para identificar al terminal, similar a la MAC Address de TCP/IP, se utiliza el siguiente valor:
    - **PEI -** es el número que identifica físicamente al terminal. Dependiendo del tipo de acceso que el UE tenga a la red puede ser diferente, aunque los valores más relevantes son los siguientes:

    - **IMEI -** que a su vez se divide en cuatro partes: TAC (Tracking Area Code), FAC (Final Assembly Code), SNR (Serial Number) y Sp (Spare)
    - **IMEISV** – que es igual que el IMEI salvo que se añaden dos dígitos finales pertenecientes al Software/Firmware del dispositivo.
- **UICC** – Conocida como tarjeta SIM, es donde se almacena la información más importante del usuario.

A medida que las diferentes tecnologías iban avanzando, las tarjetas SIM evolucionaron. En la tecnología 4G la tarjeta SIM almacena una aplicación conocida como USIM donde se guardan datos relevantes de la identidad del usuario, proporcionando una mayor seguridad que sus predecesores.

En todas las tecnologías predecesoras al 5G, el número que identifica inequívocamente al abonado, permitiendo así el *Roaming*, es el IMSI. Este está compuesto por los siguientes parámetros:

- **MCC** – Número de 3 dígitos que identifica el país de domicilio del usuario
- **MNC** – Número de 2 a 3 dígitos que identifica el operador del usuario. En España las principales operadoras con sus respectivos MNCs se pueden comprobar en la Figura 3.
- **MSIN** – Número de identificación del usuario dentro de la red

MCC	MNC	Operadora
214	1	Vodafone
214	3	Orange
214	4	Yoigo
214	7	Telefónica Móviles España (Movistar)

Figura 3: MNC España

En la tecnología 5G, sin embargo, se introduce una nueva identidad: el SUPI.

El SUPI puede tener dos valores:

- **IMSI** – al igual que las anteriores tecnologías.
- **NAI** – este valor tiene la forma de *username@realm*, definido en la especificación **RFC-4282** [2]

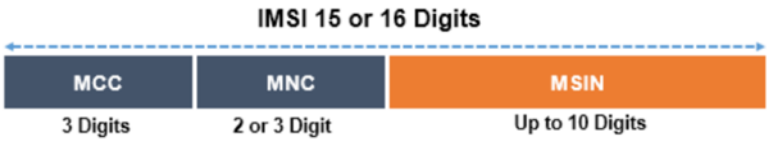


Figura 4: SUPI [3]

Por tanto, el UE se puede identificar con estos dos valores, que le identifican de forma única.

Otros parámetros relevantes dentro de la tarjeta SIM serían los siguientes:

- **ICCID** – es el número de serie de la tarjeta. Se podría comparar con el número MAC de un dispositivo en redes TCP/IP.
- **LAI** – es el número que identifica el lugar en el que se encuentra el UE, que se va modificando a medida que este se vaya moviendo.
- **MSISDN** – es el número de teléfono del dispositivo que este compuesto a su vez por varias partes: el CC (Country Code), el NDC (National Destination Code) y el SN (Subscriber Number).

Desde la tecnología 2G, se enviaba el IMSI en texto claro cuando un UE se intentaba acoplar a la red. Por ello, para mantener la seguridad y confidencialidad del IMSI, se empezaron a utilizar los identificadores temporales, un número no permanente que se va modificando a lo largo del tiempo. Cuando un usuario se conecta por primera vez a una red, si que es necesario enviar su IMSI puesto que se tiene que identificar de alguna forma. Sin embargo, el core le proporciona un número temporal para los futuros procedimientos de acople. La red recordará el ese número temporal proporcionado al UE, y podrá ir modificándolo a medida que pasa el tiempo, permitiendo así que el número IMSI sea transferido en texto claro el menor número posible de veces. Sin embargo, esto no proporciona una confidencialidad total del usuario, puesto que existen escenarios en los que hay que enviar el IMSI en texto claro. Por ello, en 5G se ha intentado remediar este problema, solución detallada en la **Sección 2.4.4**.

2.2.1.2. RAN

La RAN es la parte de la red móvil que se comunica con los clientes a través de la interfaz radio (aire), siendo el intermediario entre estos y el core de la red. Los componentes principales de la RAN incluyen estaciones bases y antenas que dan cobertura a una región o celda.

Las estaciones base en 5G se conocen como **gNodeB – gNB**. En la dirección de bajada, es decir, cuando el core de la red se quiere comunicar con el UE, estas estaciones base generan los fragmentos necesarios y los programan para su respectiva transmisión por la interfaz de aire. Por otra parte, cuando el UE quiere tener acceso a Internet, convierten los segmentos de la capa física en fragmentos IP y los redireccionan al plano de usuario del core de la red.

Estas estaciones base se comunican mediante interfaces con los siguientes nodos, como se muestra en la Figura 5:
- Core de la red (AMF/UPF): Interfaz NG-C/U
- Otras estaciones base (gNBs): Interfaz Xn
- Abonados a la red (UE): Interfaz Uu

Cada interfaz utiliza unos protocolos concretos que se pueden comprobar en la Figura 12.

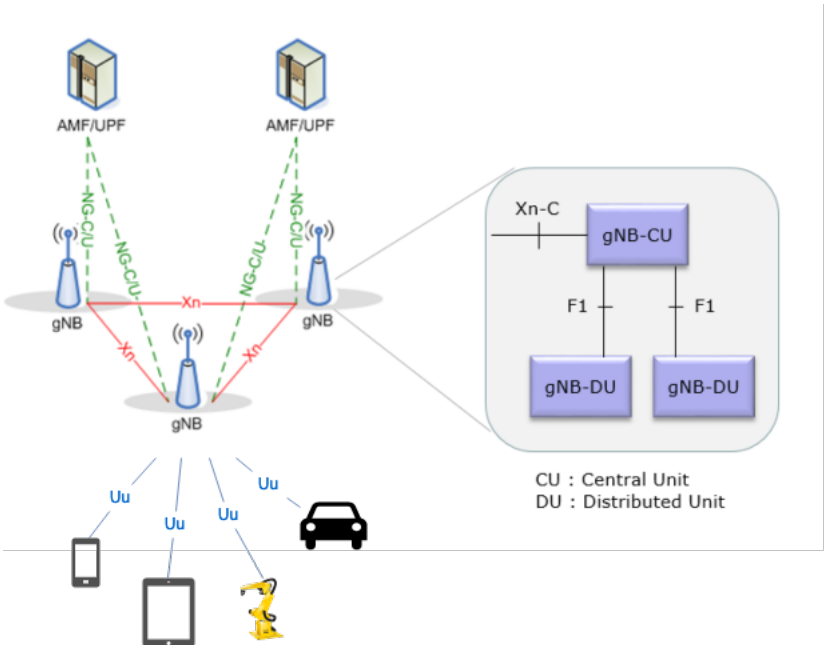


Figura 5: Arquitectura RAN 5G [4]

3GPP especifica las fases en el procesamiento de los paquetes que se efectúa en el nodo radio. Cada una de las fases ocurre en una capa del siguiente conjunto, denominado pila de protocolos:

- **RRC** – Se ocupa de la capa de control, es decir, el establecimiento de conexión, configuración de seguridad y movilidad entre muchas otras.
- **PDCP** – Es la parte encargada de procesar los datagramas IP entre los que destaca las funciones de compresión de estos datagramas, del cifrado y la integridad del mensaje
- **RLC** – Es la parte encargada de la segmentación y reensamblaje de los datagramas.
- **MAC** – Se encarga de gestionar los buffers, la multiplexación y demultiplexación
- **PHY** – La capa física es la encargada de la modulación

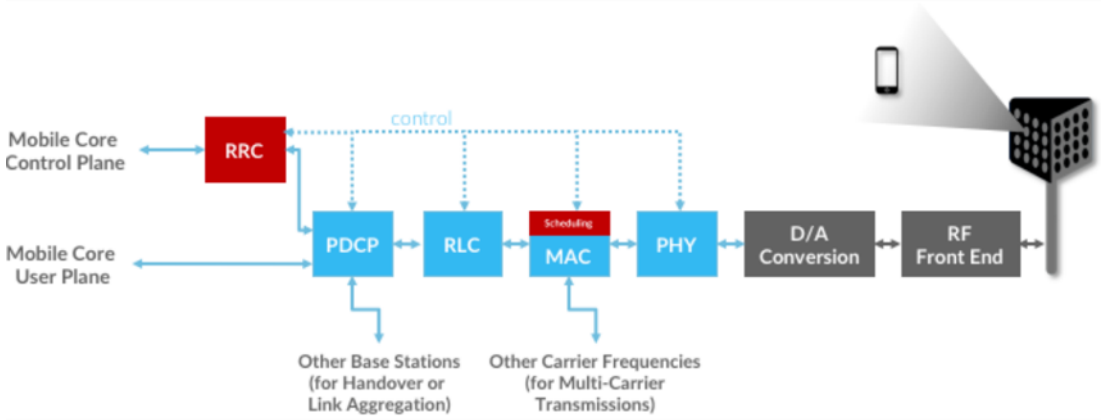


Figura 6: RAN [5]

Además, se puede observar en la Figura 5 como el gNB está dividido en dos componentes principales: gNB-CU (gNB-Central Unit) y gNB-DU (Distributed Unit), conectados mediante la interfaz F1. Esta división se debe a la posibilidad de virtualización de la capa de control, centralizando muchas de las funcionalidades de la estación base en un servidor, encargase de varios nodos de datos a la vez. Además, este nodo de control podrá colocarse en el mismo cluster que el core de la red, proporcionando una mayor flexibilidad, menor complejidad y reducción de costes.

Aunque la división entre la Unidad Central y la Unidad Distribuida se podría efectuar en distintas partes de la pila de protocolos, el estándar especifica que esta división se efectué después de PDCP como se puede observar en la Figura 7.

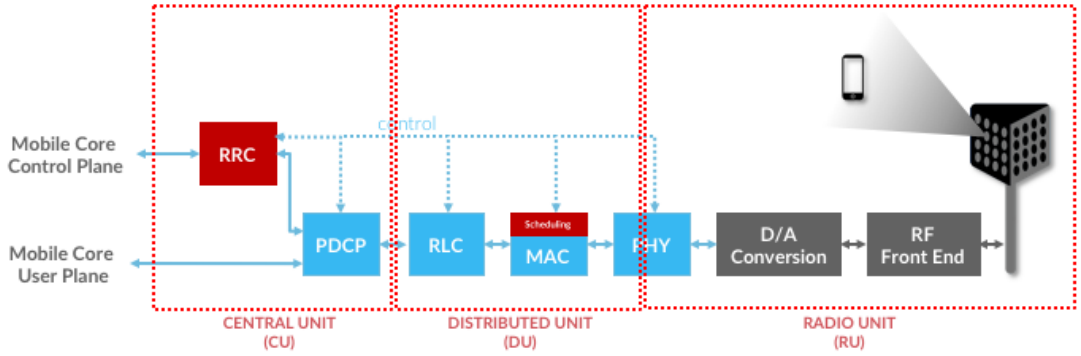


Figura 7: Split-RAN [5]

La virtualización también va a encontrar su hueco en la parte radio; implementando la arquitectura Software Defined RAN (SD-RAN). La modificación principal si se aplica la virtualización es que el RRC se dividirá en dos componentes como se puede observar en la Figura 8. El componente de la izquierda será el encargado de comunicarse con el core de la red mientras que el componente de la derecha se implementará vía software.

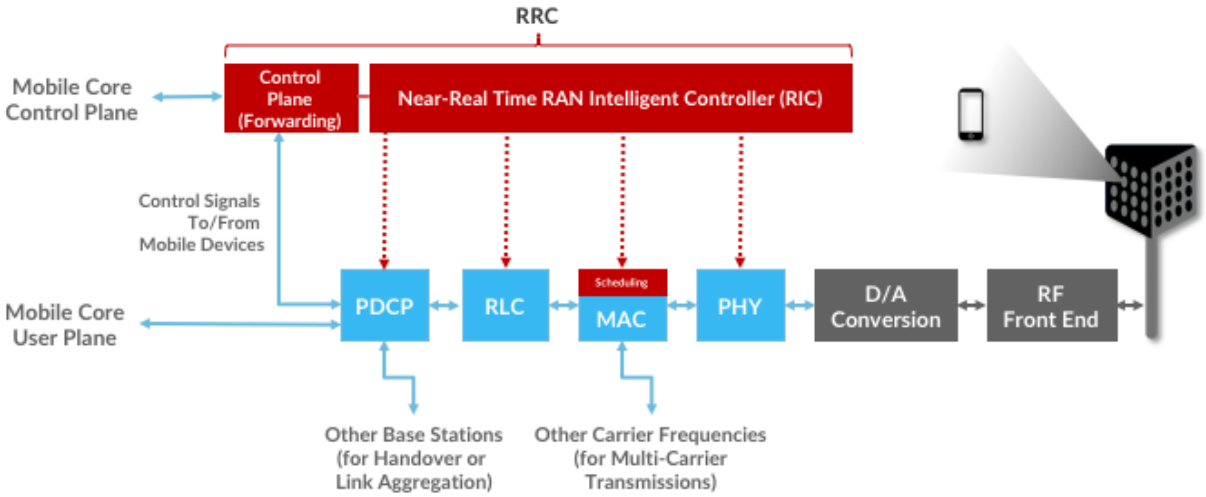


Figura 8: SD-RAN [5]

2.2.1.3. MC

El core de la red es la parte de central de una red que proporciona servicios a los usuarios entre los que destacan el enrutamiento del tráfico, la autenticación o la tarificación. En 4G se denominó EPC y en 5G se pasó a llamar NG-Core (Next Generation Core). La virtualización tendrá un papel significativo en el desarrollo de esta infraestructura puesto que muchas de las funciones de red se podrán desplegar en servidores dedicados mediante software.

Algunas de las funciones más importantes realizadas en el core de una red móvil son las siguientes:

- Enrutamiento del tráfico de red
- Conexión a una red de datos externa como Internet
- Políticas de tarificación
- Gestión de la movilidad del usuario
- Llamadas telefónicas
- Roaming
- Proporcionar seguridad mediante mecanismos como identificación y autenticación del cliente o cifrado de las comunicaciones

Aunque se han añadido nuevas funcionalidades en las redes 5G, hay ciertos nodos que tienen que existir en todo despliegue:

- **AMF** – Se podría comparar con el MME de 4G – LTE, es decir, el cerebro de la red. Se encarga de la autenticación de los usuarios, de la movilidad y servicios de localización. A diferencia del MME, el AMF no se encarga de gestionar las sesiones de los usuarios, reenviando esta información al SMF.
- **SMF** – Se encarga de la sesión de cada UE asignando las direcciones IP, de las funcionalidades Roaming o funciones DHCP.
- **UPF** – Se encarga del reenvío de paquetes entre el nodo radio e Internet, QoS del plano usuario o informa del uso del tráfico.
- **UDM** – Gestiona la identidad de los usuarios, como la generación de las credenciales de autenticación, compartiendo así parte de las funciones que tenia el nodo HSS en 4G – LTE o los permisos a la red.
- **AUSF** – Se podría comparar con el HSS de 4G – LTE, es decir, una base de datos que actúa como servidor de autenticación. Procesando las credenciales generadas por el UDM, autentica al usuario que se esta intentando registrar.
- **UDR** – Base de datos encargada de almacenar varios tipos de datos entre los que destacan los de suscripción del usuario y sus políticas de tarificación. Esta función ofrece sus datos a otras funciones como el UDM o el PCF.

Otros nodos relevantes del core de la red son los siguientes:

- **PCF** – Gestiona las reglas que otras funciones deben de cumplir, correspondiéndose con el PCRF de 4G – LTE.
- **SDSF** – Servicio de ayuda para almacenar datos estructurados

- **UDSF** – Servicio de ayuda para almacenar datos no estructurados
- **NEF** – Servidor encargado de almacenar información de otras funciones, e interactuar con terceros.
- **NRF** – Función encargada de descubrir servicios disponibles
- **NSSF** – Función que permite la selección de diferentes segmentos de red para los UEs

Como se puede observar, muchas de las funciones previamente mencionadas tienen su nodo correspondiente en 4G-LTE, puesto que el core tiene que satisfacer las necesidades básicas de toda red, es decir: comunicación con el UE, movilidad, almacenamiento de las credenciales de los usuarios, seguridad, etc. Sin embargo, aparecen muchas funciones nuevas para permitir los nuevos paradigmas como pueden ser network slicing o una arquitectura cloud. En está sección no se han mencionado todas las funciones previstas por 3GPP puesto que sería entrar demasiado en detalle, sin embargo, mencionar que se definen muchos más nodos con funciones muy concretas.

Estas funciones se comunicarán entre ellas utilizando interfaces HTTP REST, unas normas que definen como interactúan las páginas web mediante el uso de APIs. Las peticiones más importantes de este paradigma son las siguientes:

- GET – obtener información de un servidor sin modificar los datos
- POST – enviar información a un servidor
- PUT – modificar información existente de un servidor
- DELETE – borrar información de un servidor

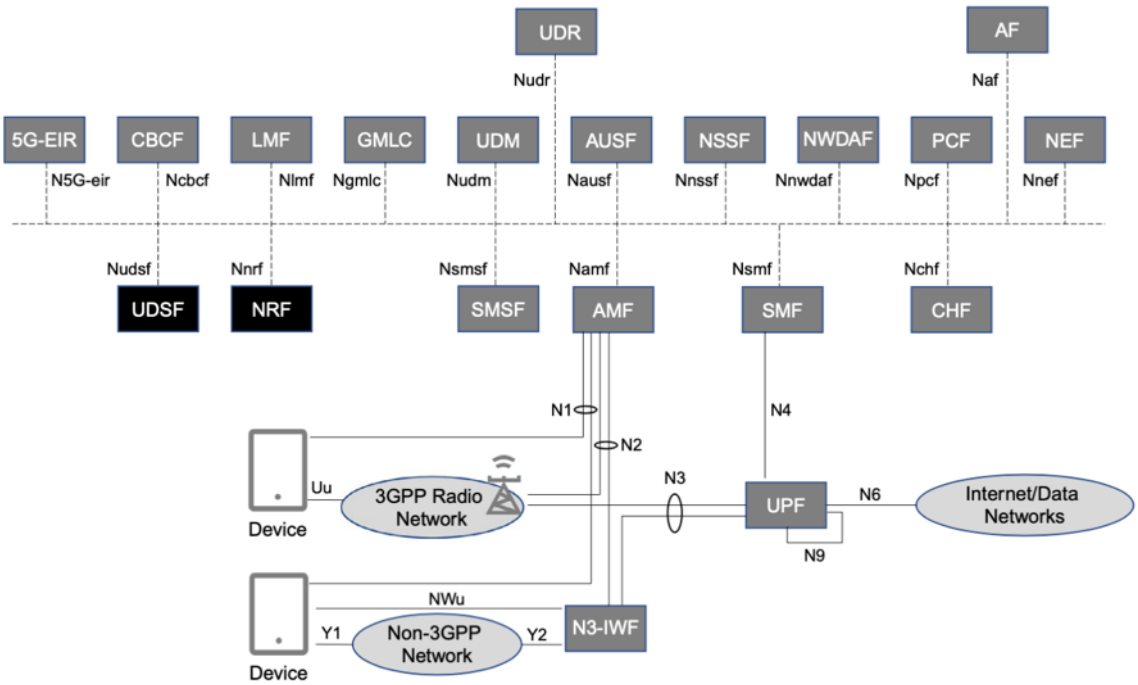
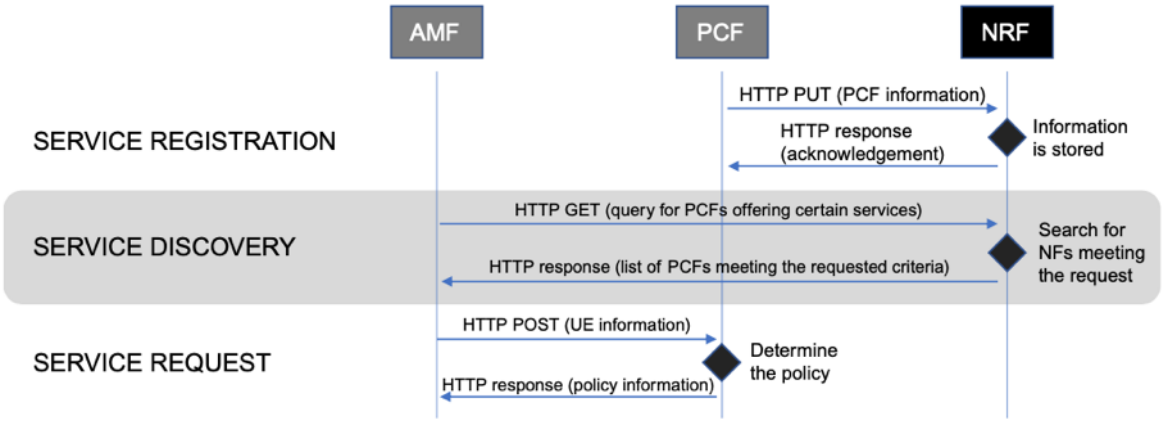


Figura 9: 5G Mobile Core [6]

Cuando dos funciones de red se comunican entre ellas, asumen un papel concreto. La función de red que envía una petición se denomina ***Service Consumer*** mientras que la función de red que recibe esa petición y realiza un servicio se conoce como ***Service Producer***.

Sin embargo, la función de red que va a solicitar información tiene que identificar los servicios disponibles contactando con el ***Service Producer*** adecuado. Esta identificación o búsqueda se denomina ***Service Discovery***. Para que esta estructura funcione, existe una función – NRF – que registra a las entidades que son ***Service Producers*** y los servicios que ofrecen, pero para ello, estas entidades necesitarán registrarse mediante un ***Service Registration.*** En el siguiente ejemplo se muestra un procedimiento de registro y búsqueda de servicio:



En la Figura 10 se pueden observar las interfaces que interconectan al UE y la RAN con el core.

- N1: Interfaz lógica entre el UE y el AMF. Un UE puede estar conectado únicamente a un AMF salvo ciertos casos de Roaming. En la imagen aparece una conexión directa con el AMF, sin embargo, esta es únicamente lógica, pues pasará por el nodo radio que reenviará la información al nodo correspondiente.
- N2: Interfaz lógica entre el gNB y el AMF. Los protocolos utilizados en esta interfaz se pueden comprobar en la Figura 12. Un gNB puede estar conectado a múltiples AMFs
- N3: Interfaz lógica entre el gNB y el UPF

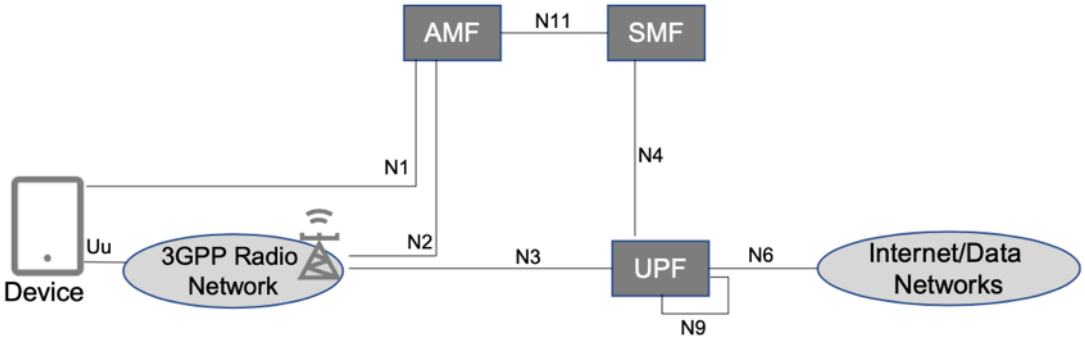


Figura 10: Interconexión entre 5G-Core y RAN

2.2.2. Despliegues

Con la existencia de la infraestructura 4G desplegada de forma global, es necesario tener en cuenta la problemática de la transición al 5G. 3GPP plantea múltiples escenarios de despliegue de una red 5G que se engloban en dos categorías principales: NSA y SA. Respecto a la Figura 11, las opciones 1, 2 y 5 pertenecen al despliegue SA mientras que las opciones 3, 4 y 7 pertenecen al despliegue NSA.

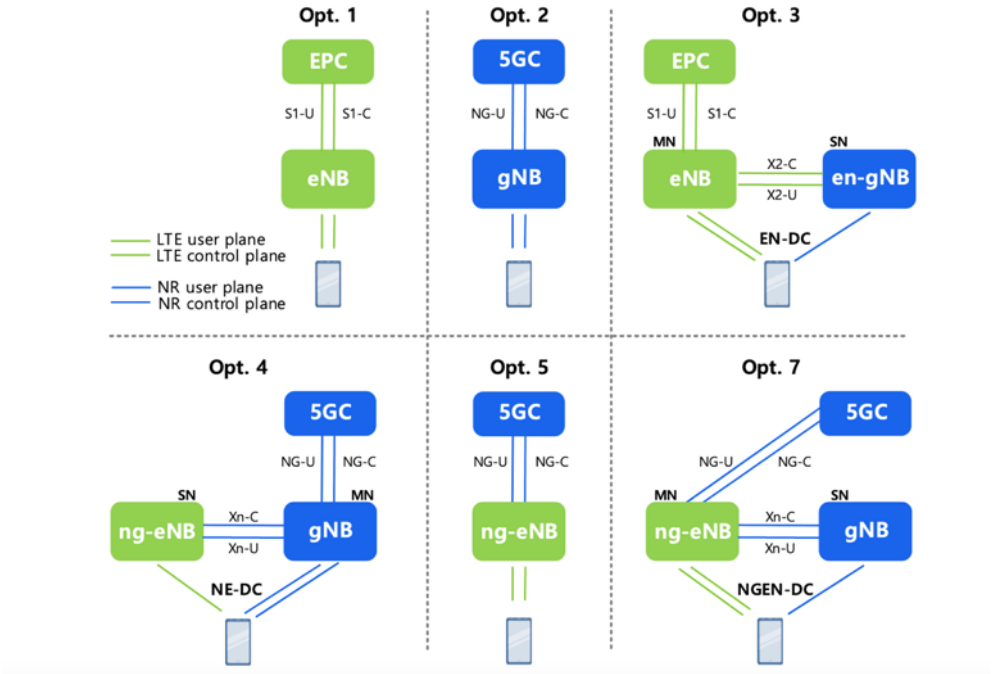


Figura 11: Tipos de despliegues de redes 5G [7]

2.2.2.1. Non-StandAlone (NSA)

Este despliegue permite el 5G apoyándose en la ya existente infraestructura 4G-LTE. Puesto que se apoya en la tecnología anterior, el UE puede conectarse tanto al nodo radio eNB como el gNB. Entre estos dos nodos, uno de ellos actúa como el nodo maestro, mientras que el otro se denomina nodo secundario.

- *Opción 3:* Como se puede observar en la Figura 11, el nodo gNB esta conectado al nodo eNB, que en este caso sería el maestro, y este último está conectado al core de la red, perteneciente a la infraestructura 4G. Esta opción es preferible para las operadoras que tienen un despliegue muy amplio de 4G a nivel nacional puesto que permite una migración sencilla a las características del 5G instalando únicamente los nodos radio. Sin embargo, como desventaja, solo se pueden aprovechar las nuevas capacidades RAN que proporciona el gNB.
- *Opción 4:* En este caso, a diferencia de la Opción 3, el nodo maestro sería el gNB que está conectado al core 5G. En esta opción, el nodo eNB tiene actualizarse a ng-eNB para poder interactuar con el nuevo nodo radio.
- *Opción 7:* Esta opción es idéntica a la Opción 4 salvo que ahora, el nodo maestro es el ng-eNB puesto que es el que está conectado al core de la red.

2.2.2.2. StandAlone (SA)

Finalmente, y mucho más reciente, se definen las redes 5G-SA. Como su nombre indica, este tipo de red será completamente 5G. Por ello, tanto la infraestructura radio como el core de la red tendrá que ser implementada. Hoy en día es prácticamente imposible encontrar alguna red comercial 5G-SA, puesto que todavía se están acabando de definir parte de los estándares, aunque si es cierto que se han desarrollado algunas redes piloto. Respecto a los tipos de despliegues, encontramos las siguientes opciones:

- *Opción 2:* Esta opción es completamente 5G, puesto que no aparece ningún nodo perteneciente a la anterior tecnología. Como se puede observar, solo existe el nodo radio, gNB que está conectado al nuevo core, el NG-MC.
- *Opción 5:* En esta opción, el nodo mejorado ng-eNB es el que está conectado el NG-MC. Sin embargo, no existe nodo gNB. Esto permite utilizar las nuevas funciones del core de la red, pero no se implementan las capacidades del nodo radio, como el uso de frecuencias milimétricas.

2.2.3. Pila de protocolos

En la Figura 12 se pueden observar los protocolos utilizados entre los distintos nodos que conforman la red tanto en la capa de control como en la capa de datos. Un protocolo es un *idioma* que define el formato y el orden en la comunicación entre dos entidades. Muchos de los protocolos utilizados en las redes 5G son los mismos que su predecesora, salvo algunos cambios en la nomenclatura o alguna actualización. Destacan dos protocolos nuevos, el SDAP entre el UE y el gNB y HTTP, protocolo muy común en las páginas web que se utilizará para las comunicaciones entre los nodos del core de la red.

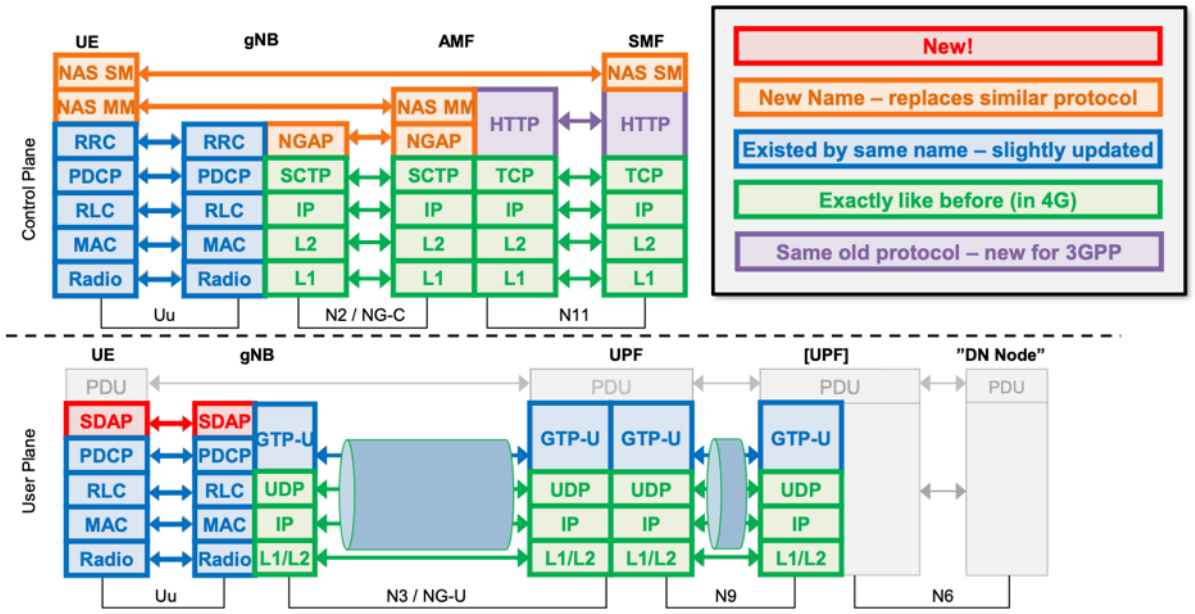


Figura 12: Pila de protocolos Plano Usuario y Control [8]

2.3. Virtualización

2.3.1. ¿Qué es la virtualización?

La virtualización hace referencia a ejecutar en software una función que tradicionalmente se ejecutaba en hardware dedicado.

Las redes son cada vez más extensas y complejas, con más usuarios conectándose al mismo tiempo y saturando el tráfico en muchas ocasiones. Los métodos tradicionales de implantación de una red móvil tienen que modificarse, dando un nuevo enfoque, y es aquí donde entra la virtualización de red. La virtualización permitirá a las redes no solo conseguir una mayor flexibilidad si no también la posibilidad de escalarlas de forma dinámica. Dos de los conceptos relacionados con la virtualización que han revolucionado muchos aspectos de las redes 5G son SDN y NFV que se explicarán a continuación.

2.3.2. SDN

El paradigma de las redes definidas por software se apoya en una arquitectura donde se utilizan controladores centralizados basados en software para gestionar el tráfico de la red apoyándose en un hardware subyacente. [9] Esto permite desacoplar el hardware dedicado del flujo de control (algoritmos de enrutamiento, configuración) flexibilizando la monitorización de los recursos, control central o control de flujos dinámico. El concepto clave de SDN es el desacoplamiento de la capa de control del plano de datos, observado en la Figura 13.

Esta nueva forma de gestionar las redes surgió de la dificultad de escalar las mismas debido a la dependencia del hardware que existía. Con el aumento

constante de los usuarios en la red, la gestión del tráfico se complicaba cada vez más, siendo poco sostenible a largo plazo.

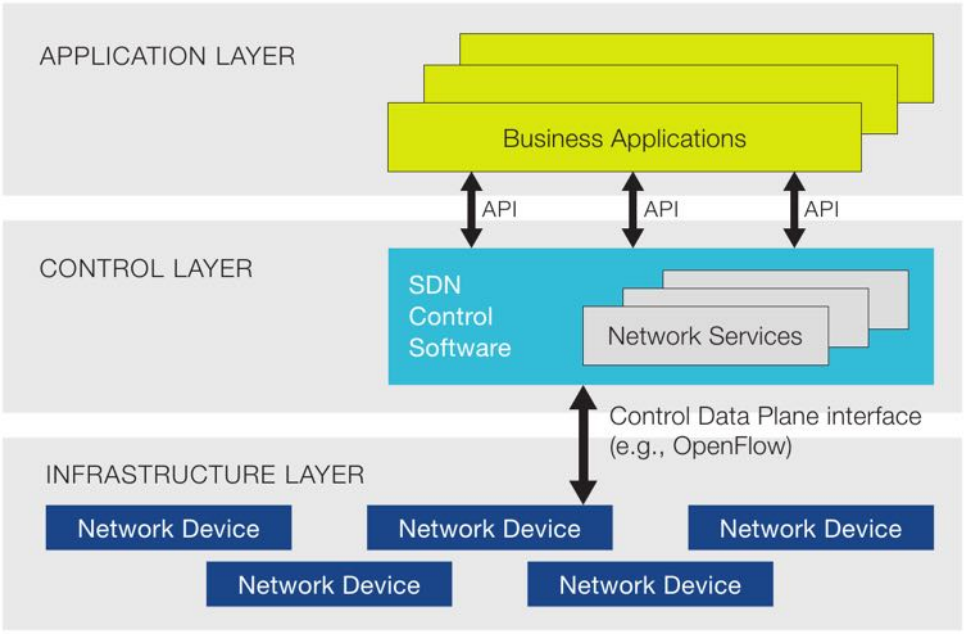


Figura 13: Arquitectura SDN [10]

2.3.3. NFV

La función principal de la arquitectura NFV es abstraer funciones concretas de la red del hardware en el que se están ejecutando. Por lo tanto, se pueden sustituir nodos de red por componentes de software que se ejecutan en contenedores o maquinas virtuales.

Este nuevo paradigma surgió de los altos costes de fabricación de los dispositivos hardware dedicados, de su complejidad y corto ciclo de vida de estos. Esto permite una reducción drástica de los costes, despliegues mucho más rápidos e incluso ahorro energético, mejorando la flexibilidad y la posibilidad de escalar la infraestructura de estas redes.

El ETSI (Instituto Europeo de Normas de Telecomunicaciones) propuso una arquitectura de NFV donde se plantea un esqueleto para su implementación. Los componentes principales de una arquitectura NFV son los siguientes:

- **VNF** - *Virtual Network Function*: Su objetivo es virtualizar un elemento de la red. Por ejemplo, virtualizar un router, que pasaría a ser un *Router VNF*. Los distintos nodos de la red se pueden virtualizar y se convertirían en VNFs. Por lo tanto, una VNF es la aplicación de software que ha virtualizado el hardware dedicado.

- **EMS** - *Element Management System:* Su encarga de gestionar un VNF, sus respectivos elementos físicos de red y proporciona funciones FCAPS (Fault, Configuration, Accounting, Performance and Security).
- **VNFM** - *Virtual Network Function Manager:* Es el encargado de gestionar uno o varios VNFs, es decir, configura, mantiene, inicializa y apaga los VNFs a su cargo.
- **NFVI** - *Network Function Virtualization Infrastructure:* Es el entorno que alberga todos los VNFs, encargado además de la gestión de memoria, almacenamiento e interconexión tanto física como virtual.
- **VIM** - *Virtual Infrastructure Manager:* Es el encargado de la gestión de los recursos del NFVI,
- **NFVO** - *Network Function Virtualization Orchestrator:* Esta compuesto por dos funciones principales: Resource Orchestration (RO) y Network Service Orchestrator (NSO). El primero gestiona los servicos de los distintos VIMs mientras que el segundo gestiona los ciclos de vida de los distintos servicios de red, utilizando los recursos provisionados por el RO.
- **OSS/BSS:** Es un conjunto de sistemas que gestionan y provisionan los servicios de red.

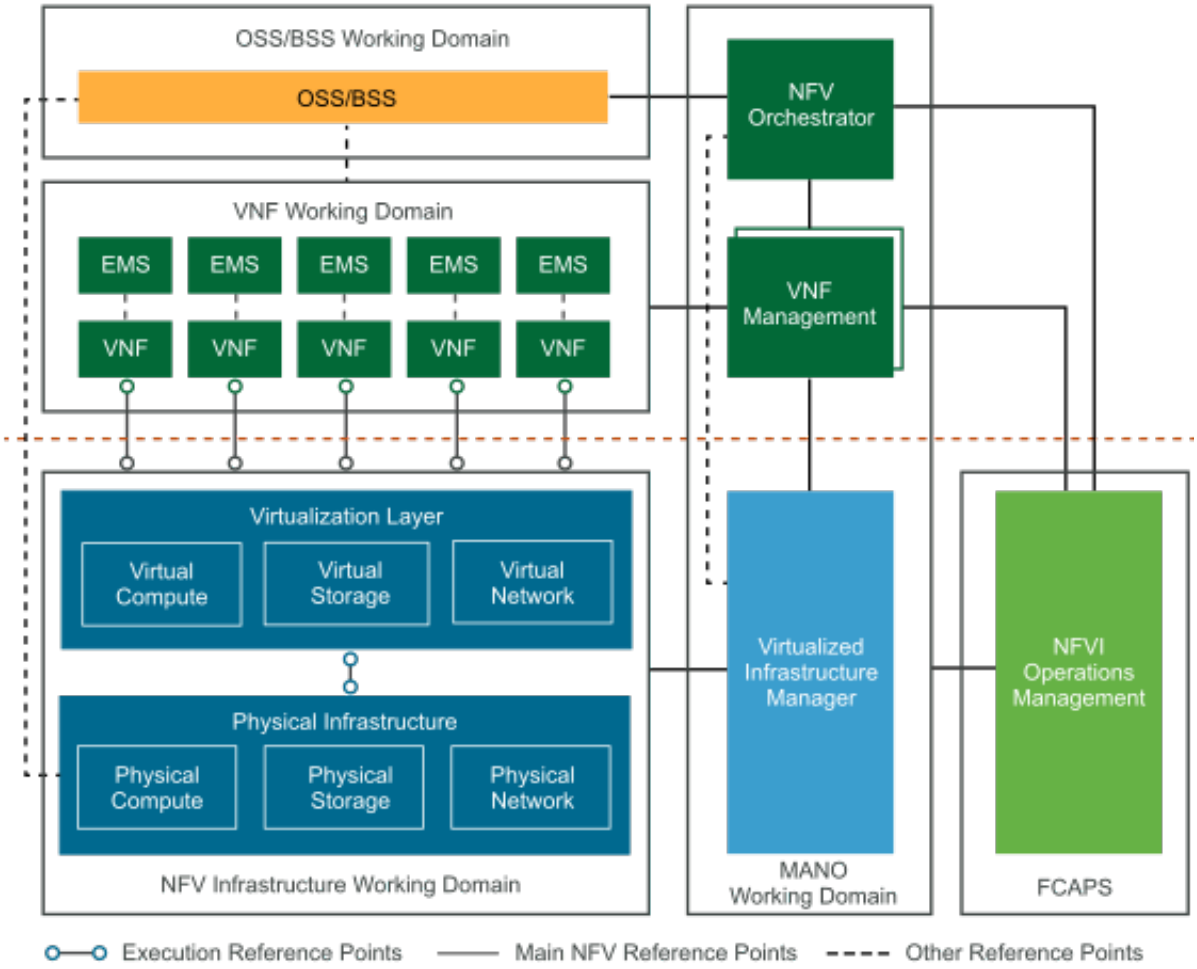


Figura 14: Arquitectura NFV [11]

Actualmente NFV se apoya en NFVI, donde un hipervisor se encarga de proporcionar Virtual Computing, Virtual Storage y Virtual Networking.

En la **Sección 2.2** se han descrito los distintos nodos del NG-MC de la red 5G. Como se puede observar, estos nodos también se denominan funciones. Es aquí donde entra la arquitectura NFV, dónde todas estas funciones se podrán virtualizar de forma que se controlen mediante software. Esto proporcionará una arquitectura de microservicios donde estas funciones podrán comunicarse entre ellas desde una infraestructura física.

2.3.4. Network Slicing

La aparición de nuevas necesidades mas allá de las que aparecen en redes móviles tradicionales, por ejemplo, dispositivos IoT o servicios que demandan una latencia mínima, enfatiza la necesidad de una arquitectura que pueda diferenciar el nivel de servicio requerido para los distintos usuarios y sus aplicaciones. Esto es lo que se conoce en 5G como *Network Slicing*.

Se define como *network slice* a una red lógica que se ejecuta sobre una red física o virtual, aislada y con un control y gestión independiente que se puede generar bajo demanda.

Esta forma de arquitectura permitirá a las operadoras:

- Desplegar múltiples funciones independientes y aisladas en una misma red
- Generar múltiples configuraciones tanto del core de la red como del RAN, por ejemplo, una para soportar dispositivos IoT y otra para un ancho de banda muy elevado
- Elegir una configuración concreta en la red dependiendo del cliente que este conectado y las aplicaciones que se estén ejecutando [12]

En la Figura 15 se pueden observar distintos *network slices* para posibles casos de uso como el IoT o una arquitectura para la sanidad. En ella se puede comprobar como existe una infraestructura física común a todos los segmentos, pero en cada ellos se gestionan o activan los nodos correspondientes, ejecutándose de forma simultanea para cubrir sus respectivas necesidades.

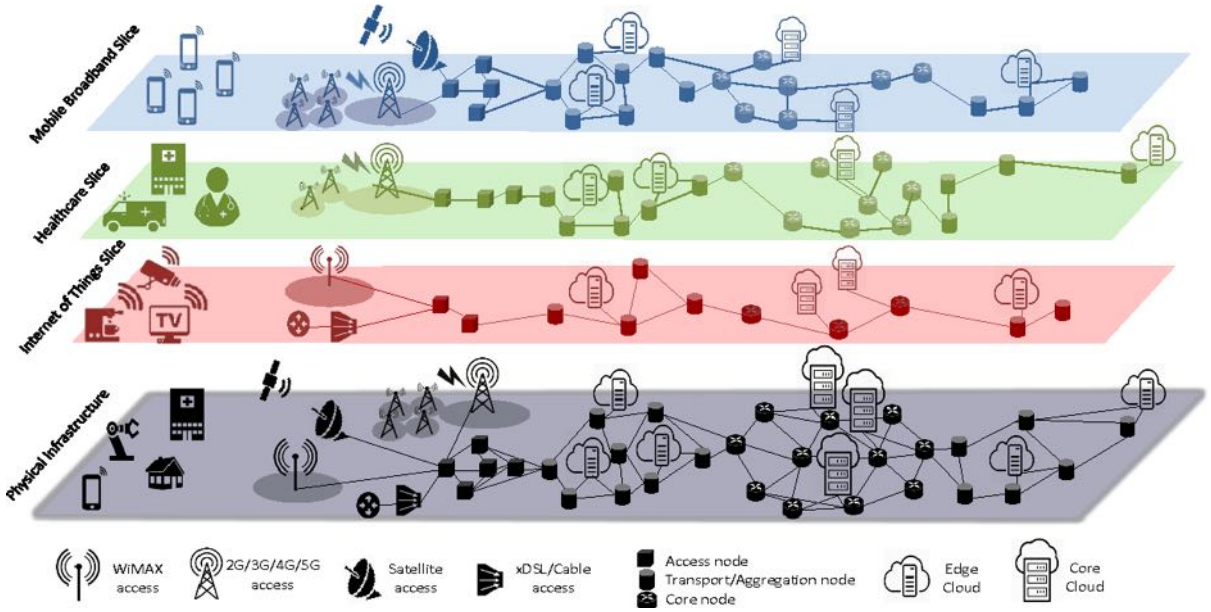


Figura 15: Segmentos de red [13]

2.4. Seguridad en redes móviles

La arquitectura de las redes móviles, como se ha mencionado anteriormente, es muy compleja, en la que forman parte infinidad de nodos, con sus respectivos protocolos e interfaces. Existiendo tantas variables, no es de asombrar que a su vez aparezcan muchas vulnerabilidades asociadas. A lo largo de las décadas, y del desarrollo de nuevas tecnologías, se han podido mitigar muchas de las vulnerabilidades críticas, proporcionando un ecosistema seguro y que ofrezca fiabilidad. Muchas de estas mejoras se deben a que los algoritmos previamente utilizados se han visto comprometidos y no ha habido más remedio que modificarlos o sustituirlos. Sin embargo, todavía existe mucha superficie de ataque y con el incremento sin precedentes de los dispositivos conectados esta seguirá aumentando.

Los aspectos de seguridad principales de una red son tres: la privacidad del cliente, autenticación del usuario y confidencialidad en las comunicaciones. Estas características son básicas para que una red permanezca segura, por lo que se explicarán a continuación. Es importante señalar que serán necesarios muchos más diseños de seguridad para proporcionar una red robusta ante distintos ataques.

2.4.1. Identificación y Autenticación

Es de vital importancia que la red se asegure de identificar al usuario puesto que evita posibles fraudes, por ello, se utiliza el número IMSI como identidad privada del abonado. Sin embargo, como este número identifica de forma única al UE, obtenerlo significaría no solo comprometer la posición geográfica de una tarjeta

SIM, si no también se podrían perpetrar numerosos ataques que vulnerarían la privacidad del usuario. Por ello, será necesario transmitirlo en el menor número de veces posible en texto claro en canales no seguros (el aire). Una de las soluciones propuesta por los estándares desde la segunda generación es el uso de una identidad temporal.

Para poder utilizar este nuevo número temporal, la base de datos de la red debe tener una entrada con la relación entre el IMSI y esta identidad temporal. La identidad temporal se generará en distintos procedimientos, como el de actualización de ubicación y la red enviará este número al UE para que lo almacene. Sin embargo, si el UE no ha recibido previamente un identificador temporal por parte de la red tendrá que identificarse con el IMSI. Esto ocurre por ejemplo cuando un UE se conecta a una red por primera vez.

El proceso de identificación sigue el siguiente patrón general:

- Después de que el UE y la estación base se sincronicen entre ellos para poder comunicarse, el UE envía una solicitud de conexión, donde en general, envía su identidad privada o temporal
- Si es la primera vez que el UE se conecta a la red, enviará el IMSI por lo que ya se ha identificado y se prosigue con el proceso de autenticación
- Si el UE envía un identificador temporal, la red comprueba ese número y si existe algún fallo, como por ejemplo un desconocimiento del identificador, le enviará al UE una petición de identificación
- El UE contestará con su identidad privada, es decir el IMSI y se proseguirá con el proceso de autenticación

A partir de la identificación del usuario, puede comenzar el proceso de autenticación. Este proceso sigue un patrón común en las distintas tecnologías móviles, conocido como *Challenge:*

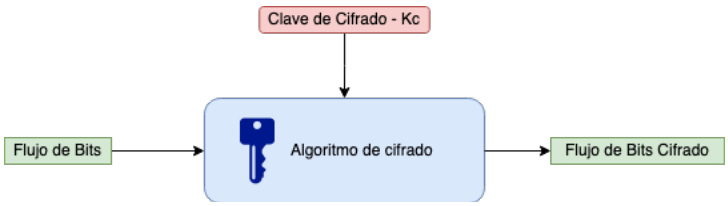
- La red genera una secuencia aleatoria para el cálculo de unos parámetros que envía al UE
- Tanto el UE como la red generan unos vectores de autenticación en base a esa secuencia con el uso de unos algoritmos preestablecidos
- Los vectores generados se comparan:
    - Si son iguales, el UE se ha autenticado correctamente
    - Si son diferentes la autenticación ha fallado
- A partir de aquí se podrán cifrar las comunicaciones mediante un algoritmo de cifrado conocido por ambos

Es importante recalcar que en 2G la red era la que autenticaba al UE y no viceversa. Esto generaba la posibilidad de perpetrar distintos ataques como las estaciones falsas. Por ello, en las siguientes tecnologías decidieron implementar la autenticación de la red por parte de la estación móvil.

2.4.2. Confidencialidad

Para asegurar la confidencialidad en las comunicaciones entre el usuario y la red es necesario cifrar los canales por los que va a circular el tráfico. El proceso de cifrado del canal de comunicación comienza después de la autenticación del usuario. Para cifrar los canales son necesarios los siguientes aspectos:

- Establecimiento de la clave – se establece una clave de cifrado la cual utilizarán tanto red como el usuario para comunicarse de forma segura. La clave de cifrado utilizada no deberá enviarse en texto claro a través de un canal no seguro, puesto que vulneraría todas las comunicaciones entre los nodos. Esta clave se utilizará, además de otros parámetros, en un algoritmo de cifrado también común a ambos nodos.
- Inicio del proceso tanto de cifrado como descifrado
- Sincronización – El proceso de cifrado debe estar sincronizado en ambos extremos de la comunicación puesto que, si no coinciden en el mismo bit, habrá errores de lectura.
- Handover – La clave de cifrado tendrá que ser transmitida de estación base a estación base cuando exista movilidad por parte del usuario



Uno de los aspectos claves de la confidencialidad es el algoritmo de cifrado utilizado en las comunicaciones. A continuación, se presentan los algoritmos desarrollados de las distintas generaciones de redes móviles [14]:

- GSM: El cifrado del tráfico da comienzo en la estación base mediante el mensaje de *Ciphering Mode Command* donde envía al UE el algoritmo a utilizar.
  - A5/0 – No existe cifrado
  - A5/1 – Clave de 64 bits pero esta roto
  - A5/2 – Clave de 64 bits pero esta roto por lo que su uso está prohibido
  - A5/3 – Clave de 64 bits que se puede extender a 128 bits, siendo el más seguro de esta tecnología
  - A5/4 – Igual que el A5/3 pero con una clave de 128 bits
- GPRS
  - GEA0 – No existe cifrado
  - GEA1 – Clave de 64 bits pero esta roto por lo que su uso está prohibido
  - GEA2 – Clave de 64 bits
  - GEA3 – Clave de 64 bits que se puede extender a 128, similar al A5/3
- UMTS: Estos algoritmos son conocidos como f8 y comienza cuando la estación base envía el mensaje *RRC Security Mode Command*
  - UEA0 – No existe cifrado
  - UEA1 – Clave de 128 bits basado en KASUMI, similar al A5/3
  - UEA2 – Clave de 128 bits basado en SNOW 3G
- LTE: El cifrado de las comunicaciones comienza con el mensaje de *NAS Security Mode Command*

    - EEA0 – No existe cifrado
    - EEA1 – Igual a UEA2 basado en SNOW 3G
    - EEA2 – Clave de 128 bits basado en AES
    - EEA3 – Clave de 128 bits basado en ZUC
- 5G
    - NEA0 – No existe cifrado
    - NEA1 – Igual a EEA1
    - NEA2 – Igual a EEA2
    - NEA3 – Igual a EEA3

2.4.3. Estaciones falsas

Las estaciones falsas tienen como objetivo que el usuario se registre en esta estación obteniendo el control completo de las comunicaciones. Estos ataques eran muy comunes en la segunda generación de telefonía ya que el UE no autenticaba a la red, conectándose a toda estación que detectase, siendo una vulnerabilidad crítica que modificaron en las siguientes generaciones. A partir de 3G, el UE si que autenticaba a la red, por lo que sería mucho más complicado tener acceso a las comunicaciones de un dispositivo y perpetrar este ataque.

La metodología general utilizada en este ataque consta de los siguientes pasos:

- En primer lugar, obtener la identidad del abonado, es decir, el IMSI.
- Una vez se ha obtenido el IMSI del usuario, y está registrado en la estación base, se tendrá que "obligar" a la estación móvil a conectarse a su red. Para ello existen diversas opciones:
    - Utilizar un inhibidor de frecuencias para que se desconecte de su celda actual
    - Suplantar a alguna celda y emitir una señal con mayor potencia para que la estación móvil decida cambiarse
- Una vez se conecte a la estación base, se pueden rastrear las comunicaciones teniendo en cuente lo siguiente:
    - La estación base no debe cifrar los mensajes para agilizar todo el proceso de captura de las comunicaciones

2.4.4. Aspectos de seguridad en 5G

El 5G conectará a miles de dispositivos en una red muy compleja, por lo que la superficie de ataque será mucho mayor. No solo eso, si no que estas redes soportarán infraestructura crítica y un ataque a esta infraestructura podrá tener un impacto mucho mayor en la sociedad que en otras tecnologías. Por ello, es necesario hacer un especial énfasis en la investigación y destacar los retos de seguridad en este tipo de redes móviles.

Los nuevos aspectos a tener en cuenta para securizar las redes 5G serían los siguientes:

- Privacidad de los usuarios
- Seguridad de la virtualización; SDN y NFV
- Seguridad de los miles de nuevos dispositivos conectados

2.4.4.1. Privacidad de los usuarios

Uno de los grandes problemas de las generaciones anteriores al 5G eran las ya mencionadas estaciones falsas, o IMSI Catchers, puesto que la identidad del dispositivo se enviaba en texto claro. Esto es debido a que el proceso de identificación ocurría antes del proceso de autenticación. Afortunadamente, esta "vulnerabilidad" ha sido abordada en 5G.

El identificador único en la tecnología 5G se denomina SUPI, y este no se debe enviar directamente en texto claro, como se detalla en la especificación TS 33.501 de 3GPP, «SUPI should not be transferred in clear text over NG-RAN except routing information, e.g. Mobile Country Code (MCC) and Mobile Network Code (MNC)». [15] La única excepción serán las llamadas de emergencia en las que no sería necesaria la autenticación, pero en este caso se enviará en texto claro el IMEI.

Para no transferir el SUPI en texto claro se utiliza una identidad cifrada conocida como SUCI. Si la red es segura, existirá una clave pública y una clave privada de la operadora y la clave pública irá almacenada en la tarjeta SIM de fábrica del UE, por lo que se podrá enviar el SUCI de forma segura para la posterior autenticación del usuario.

El SUCI está formado por los siguientes parámetros:

- Tipo de SUPI – Valor comprendido entre 0 y 7, el cual identifica el tipo el SUPI almacenado
  - 0: IMSI
  - 1: NAI
  - 2 a 7: Valores para posibles usos en el futuro
- Home Network Identifier – Identifica la red del usuario
- Routing Indicator – 1 a 4 decimales asignados por la operadora
- Protection Scheme Identifier
  - Null – No existe ninguna protección por lo que el SUPI se transfiere en texto claro, es decir, en el Scheme Output se transfiere el MSIN
  - Profile A
  - Profile B
- Home Network Public Identifier – Consiste en una clave pública del operador. Si el Protection Scheme Identifier es nulo, este valor es 0.
- Protection Scheme Output – El valor en claro o cifrado del MSIN del SUPI

En el Protection Scheme Identifier existen dos perfiles posibles a utilizar. Ambos se basan en el Elliptic Curve Integrated Encryption Scheme (ECIES), un esquema de encriptación utilizado comúnmente en criptografía

- Profile A
  - ECC Clave Pública de 256 bits

    - Texto Cifrado
- Profile B:
    - ECC Clave Pública de 264 bits
    - Texto Cifrado

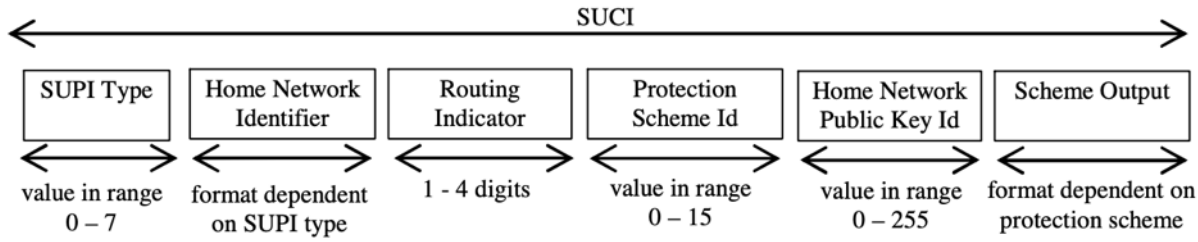


Figura 16: Estructura del SUCI [16]

Por tanto, el registro por parte del abonado se hará de la siguiente forma:

1. Registration Request por parte de la UE, en el que se enviarán en texto claro tanto el MCC como el MNC y por otra parte el SUCI o el GUTI
2. Si se envía el SUCI, la red comprobará cual es el SUPI en su base de datos
3. Se generá un GUTI y se envía al usuario
4. Se acepta el Registration Request y las siguientes veces que el usuario quiera conectarse enviará el GUTI en vez del SUCI

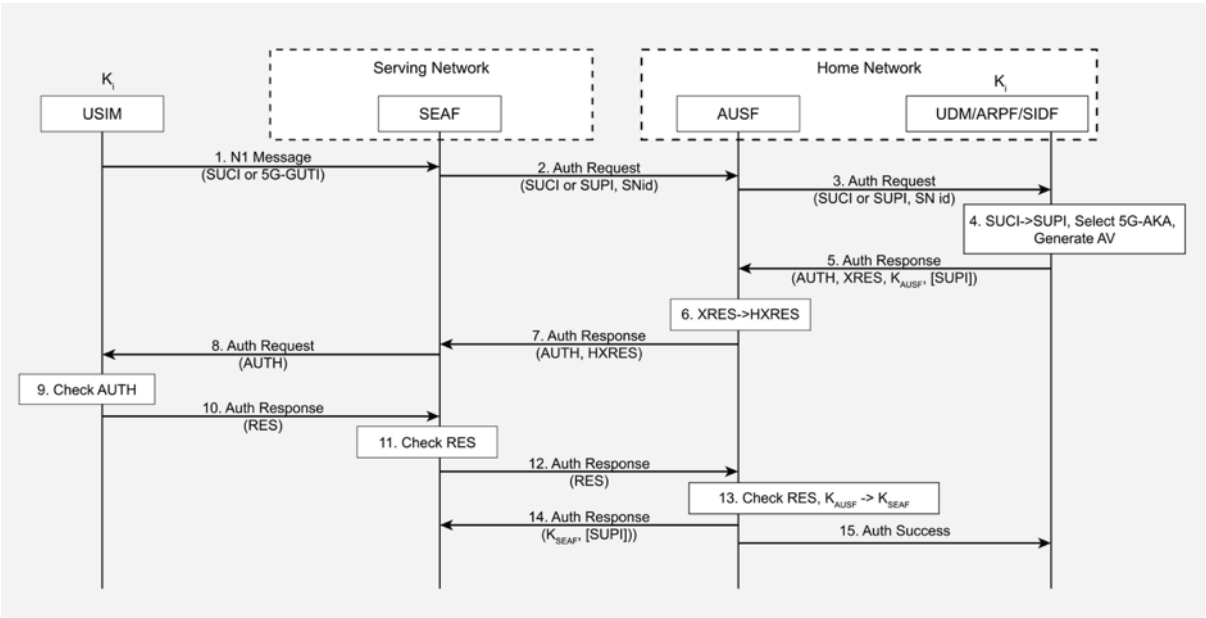


Figura 17: Registration Request 5G [17]

Además, otra característica que previene el rastreo del SUCI es que este no es estático, es decir, en cada *Identity Response* el SUCI enviado se modifica. Esto es debido a que para generar el SUCI no se utiliza únicamente la clave pública de la operadora, además, se utiliza la clave privada – ephemeral key – y se envía a través del canal la clave pública a la red.

2.4.4.2. Seguridad de la virtualización

Uno de los grandes avances y revolución de esta nueva tecnología es la virtualización, que, por una parte, aumenta la flexibilidad y escalabilidad de la red, pero por otra, ofrece una nueva superficie de ataque.

Aunque existe un amplio abanico de posibilidades de configuración y despliegues, se están generando unos estándares de seguridad aplicables a todas las situaciones, pero debido a la complejidad de estos, es probable que las operadoras acaben protegiendo su infraestructura de forma especifica a cada despliegue. Los estándares los especifica ETSI, donde define las contramedidas que hay que aplicar a los ataques de virtualización.

Los principales riesgos asociados a la virtualización descritos por la ETSI son los siguientes:

- Escapar de la maquina virtual a la maquina real, donde se podrá obtener el control de la red.
- Securización y autenticación de las APIs
- Aislamiento tanto de elementos de hardware, recursos y VNFs críticas. Si un elemento virtual de la red es atacado este aislamiento no permitirá que otros dispositivos no se vean afectados.
- En SDN en general existe un controlador por lo que tiene que existir autenticación entre el controlador y las entidades a su control puesto que, si el primero es atacado, las entidades se verían afectadas.

2.5. Procedimiento de acceso a la red 5G

Uno de los objetivos en este proyecto es comprobar, aparte de la seguridad general de las redes 5G, si el ataque conocido como IMSI Catcher sigue funcionando en esta nueva tecnología. Como su nombre indica, el fin de este ataque es obtener el número IMSI – detallado en el **Capítulo 5** - o lo que es lo mismo, la identidad del abonado. Aunque existen métodos para dificultar la obtención del IMSI, el UE tiene que identificarse de alguna forma, y esto ocurre en el procedimiento de acceso a una red 5G, por ello se ha considerado importante incluir este apartado.

2.5.1. NSA

En la **Sección 9.2.1** se representan los pasos para que un UE se conecte a una red 5G-NSA. En un primer lugar, el UE se conecta al nodo eNB, por tanto, los mensajes iniciales intercambiados entre los distintos nodos de red se basan en la tecnología 4G-LTE. Como en esta tecnología sigue existiendo la capacidad de IMSI Catching, era muy probable que se pudiese realizar el IMSI Catcher, como se podrá observar en el **Capítulo 0.** Una vez que la red acepte la conexión del UE, y este le envíe la información de sus capacidades, incluyendo que puede conectarse a una red 5G, comenzará la transición al nodo gNB. Es importante recalcar que en 5G-NSA existe una conexión dual, es decir, el UE estará conectado a la vez tanto al eNB como el gNB. Esto es debido al uso de MIMO (Multiple Input Multiple

Output). Cuando sea necesario aumentar la transmisión de datos, por ejemplo, al descargar una película, se utilizará el nodo gNB.

- En primer lugar, el UE se sincroniza con las distintas frecuencias correspondientes con la tecnología y comprueba si la celda detectada corresponde con el operador de la tarjeta SIM. Esto es posible puesto que el nodo radio envía periódicamente información de su celda mediante el *System Information Block (SIB).*
- El siguiente paso se conoce como *Random Access Procedure*, donde la red sabe que un UE quiere intentar conectarse por lo que le proporciona recursos para la comunicación inicial
- Cuando la comunicación inicial ha finalizado, le sigue el *RRC Connection Establishment*, compuesto por tres mensajes principales:
    - RRC Connection Request: en este mensaje se incluyen dos parámetros importantes
        - ue-Identity: Si el UE se ha conectado anteriormente a esta red enviará in identificador temporal conocido como S-TMSI. Si es la primera vez que se conecta enviará un número aleatorio.
        - establishmentCause: Existen múltiples causas de conexión a una red entre las que destacan la de emergencia, conexión normal o de información
    - RRC Connection Setup: contiene información acerca de los canales lógicos proporcionados por la celda al igual que información de configuración. Aparecen múltiples parámetros físicos.
    - RRC Connection Setup Complete: se informa al nodo radio la finalización de la conexión RRC, incluyendo tres mensajes
        - selectedPLMN-Idenity
        - registeredMME
        - dedicatedInfoNAS: en este mensaje es donde se incluye el Attach Request – detallado en la **Sección 2.5.1.1** – y donde se encontrará la información más vulnerable del UE.
- Una vez completado este *three-way handshake* el UE pasará al estado de RRC Connected y se enviará el Attach Request al MME.
- Se realizará la autenticación correspondiente mediante un intercambio de mensajes entre el MME y el HSS. Las bases de la autenticación se han descrito previamente en la **Sección 2.4.1.**
- Cuando todas las claves se han generado correctamente tanto de la parte del UE como de la red y coinciden, la red envía el *Attach Accept* al UE.
- UE Capability Enquiry: En el siguiente paso el eNB le pide al UE que le envía la información sobre sus capacidades
- UE Capability Information: En este mensaje se incluye toda la información necesaria de las capacidades del UE, de las cuales se puede destacar lo siguiente:
    - accessStratumRelease: Valor de las especificaciones en las que el UE ha sido creado
    - Capacidad para conectarse a 5G – en este caso el atributo rat-Type sería EUTRA-NR

    - Bandas que soporta el UE para la tecnología radio prevista en el enquiry
    - halfDuplex: Capacidad para halfDuplex

  A raíz de esta información, la red sabrá si el UE es capaz de conectarse a un nodo gNB y hará la transición correspondiente.
- A partir de aquí, hay un intercambio de Security Commands para decir al UE que active la seguridad del Access-Stratum y se finaliza el procedimiento de acople con un *Attach Complete*
- A partir de aquí el UE ya está conectado a las red 4G – LTE. Sin embargo, el objetivo es tener capacidades 5G por lo que se comenzará con la transición.
- Comprobación de sincronización 5G
    - En primer lugar, se envía la señal de sincronización el UE y se mide la calidad de la señal enviando esa información al eNB
- Adición del nodo secundario
    - Se hacen unas comprobaciones iniciales para comprobar si la distancia del usuario y el ancho de banda son suficientes para la nueva tecnología
    - Addition Request: el eNB decide añadir al gNB como nodo secundario por lo que le envía información al nuevo nodo sobre las capacidades del UE y la configuración establecida
- El eNB reenvía la reconfiguración necesaria al UE asignando los recursos de radio correspondientes al 5G
- Finalmente, se notifica al MME que se va a cambiar de nodo radio con un *Modification Indication* y el UE se conecta al nuevo nodo, sincronizándose inicialemente.
- El UE ya esta conectado al gNB por lo que tiene las nuevas capacidades de la tecnología 5G
- Cada cierto tiempo, el UE envía información al eNB de la conexión con el nodo 5G, por si en algún caso habría que volver a la conexión 4G.

2.5.1.1. Attach Request

Como se ha mencionado, en este mensaje reside información clave para que el UE se pueda conectar a la red, desde su identidad hasta información de cifrado. Podemos encontrar los siguientes valores principales:

- *EPS Attach Type:* este valor indica la finalidad de la conexión
- ***EPS Mobile Identity:*** este parámetro es el que incluirá el GUTI, IMSI o IMEI. Si ya se ha conectado a la red previamente, únicamente tendrá que enviar el GUTI. Sin embargo, en el caso de que este valor no sea válido o sea la primera vez que el UE se conecta a la red, tendrá que enviar el IMSI en texto claro. Es aquí donde se puede obtener este número, es decir, la identidad del abonado. En el caso de que el EPS Attach Type sea de emergencia, se enviará el IMEI.
- Otros parámetros relevantes son los siguientes: *UE network capability, ESM message container, Old P-TMSI signature, Additional GUTI, Last*

visited registered TAI, Old location area identification y *Voice domain preference.*

2.5.2. SA

En la **Sección 9.2.2** se describen los pasos que se siguen para realizar una conexión satisfactoria a una red 5G-SA.

- Al igual que en 5G-NSA en el primer paso, el UE busque una celda correspondiente a su operador y se sincroniza con ella para poder realizar la conexión.
- Cuando se obtiene una sincronización tanto de subida como de bajada, el UE comienza el RRC Connection Establishment
    - RRC Connection Request: este mensaje incluye dos parámetros al igual que en 5G-NSA, es decir, el ue-Idenity y el establishmentCause.
    - RRC Setup:
    - RRC Setup Complete
        - Registration Request
            - 5G-GUTI or SUCI: En este caso, en teoría no se debería enviar nunca el IMSI en texto claro, salvo que no se implementa estas nuevas características de confidencialidad.
            - Selected PLMN ID
            - Last TAI
            - UE Capability
            - List of PDU Sessions
- Se selecciona el AMF para la sesión
- Identity Request
- Identity Response: Es aquí dónde se proporciona el SUCI
- Comienza el proceso de autenticación, por lo que el AMF se empieza a comunicar con el AUSF
    - El AUSF envía el SUCI o 5G-GUTI al UDM/ARPF/SDF que serán los encargados de descifrar el SUCI, obteniendo la identidad real del usuario (SUPI), calculando los vectores de autenticación y enviándoselos al AUSF
    - El AUSF envía al AMF los resultados y envía una petición de *Authentication Request* al UE
    - El UE responde con un *Authenticaion Response* con los valores obtenidos a partir de los vectores al AMF que comprueba con las demás entidades estos valores
- Se comprueba si el UE pertenece a alguna lista del EIR. Este nodo tiene varias bases de datos donde almacena las identidades de los usuarios. Existen tres listas principales:
    - Negra – usuarios a los que no se les permite conectarse a la red
    - Gris – usuarios a los que se les permite conectarse a la red, pero pueden ser rastreados
    - Blanca - usuarios a los que se les permite conectarse a la red

- El core de la red realiza un par de funciones entre las que destaca la obtención de los datos de suscripción del usuario de la base de datos
- Después de estas funciones, el gNB y el UE intercambian mensajes de seguridad para empezar a cifrar las comunicaciones entre ellos
- Se reconfigura de la conexión RRC y finalmente el UE envía a la red la finalización del Registro a la red

3. Estado actual redes 5G

3.1. Licenciamiento de frecuencias

Existen dos bandas de frecuencias principales:

- FR1 (Frequency Range 1): 410 MHz – 7 125 MHz
- FR2 (Frequency Range 2): 24 250 MHz – 52 600 MHz

En el primer rango existen frecuencias que ya se emplean en otras tecnologías como GSM 900, DCS o LTE. Sin embargo, esta previsto que a lo largo de los próximos años un porcentaje de este ancho de banda sea utilizado por el 5G. El segundo grupo de frecuencias, FR2, es conocido como mmWave debido a su la naturaleza milimétrica. En este rango de frecuencias, la cobertura es mucho menor pero la velocidad de transmisión de datos se incrementa considerablemente.

En base al Plan Nacional de 5G propuesto por el Ministerio de Energía, Turismo y Agenda Social [18] dentro de las dos bandas de frecuencias se identifican tres rangos relevantes:

- Banda 3,4 GHz – 3,8 GHz se considerará como la principal para iniciar las comunicaciones en toda Europa.
- Banda de 694 MHz - 790 MHz como banda 5G con un rango de cobertura y penetración en las paredes superior, proporcionando una calidad de servicio mayor.
- Banda de 24,25 GHz – 27,5 GHz como el primer rango dentro de las frecuencias milimétricas en implantarse, con un rango de cobertura mucho más limitado, pero velocidades de transmisión de datos muy elevadas.

El estado actual de estos rangos es el siguiente:

- Banda 3,4 GHz – 3,8 GHz:
    - Banda 3,4 GHz – 3,6 GHz: Se subastaron al inicio del siglo para fomentar el desarrollo de tecnologías como LMDS o WiMax. Aunque fueron adjudicadas a operadoras que ya no existen, los anchos de banda pasaron a formar parte de las grandes operadoras de hoy en día. Por otra parte, en la Tabla 1 se pueden comprobar a que operadoras corresponden los distintos anchos de banda dentro de este rango. Existían 40 MHz pertenecientes al Ministerio de Defensa, y 20 MHz fueron subastados en febrero de 2021 y cayeron en manos de Movistar y Orange. Además, el rango de 3480 MHz a 3500 MHz se migrará a la parte baja del espectro y no se podrá utilizar para no interferir con las comunicaciones de la OTAN que utilizan frecuencias justo por debajo de los 3400 MHz. Finalmente, por cuestiones de eficiencia en los próximos años, esta banda se reorganizará para que las operadoras puedan utilizar portadoras con más ancho de banda
    - Banda 3,6 GHz – 3,8 GHz: Este rango fue adjudicadas en 2018 a las operadoras como se puede comprobar en la Tabla 2. Este rango se

vendió en la subasta por un precio total de 437,65 millones de euros y las operadoras serán titulares de este rango durante los siguientes 20 años, es decir, hasta 2038. Estas frecuencias ya se reorganizaron como se puede observar en la Tabla 2.

- Banda de 700 MHz: Se utilizaba para la Televisión Digital Terrestre (TDT), por lo que ha tenido que liberarse a lo largo de estos últimos años para que no haya interferencias. La TDT pasará de ocupar el rango de 700 MHz a la de 600 MHz. Esta banda se subastará en julio de 2021. Está previsto que este rango, puesto que su penetración será la más elevada de las frecuencias principales del 5G, sea la más cara de la historia. El Gobierno de España ya ha lanzado la subasta con un precio inicial de 995,5 millones de euros.
- Banda 24,25 - 27,5 GHz: Esta banda actualmente tiene disponibles tres bloques para un uso inmediato. La otra parte del espectro se utiliza para radioenlaces o sufren limitaciones.
    - 24,25 GHz – 24,661 GHz: disponible para uso inmediato, sin limitaciones
    - 24,661 GHz – 24,717 GHz: utilizado por sistemas punto-multipunto
    - 24,717 GHz – 25,445 GHz: radioenlaces
    - 25,445 GHz – 25,669 GHz: disponible para uso inmediato, sin limitaciones
    - 25,669 GHz – 25,725 GHz: utilizado por sistemas punto-multipunto
    - 25,725 GHz – 26,453 GHz: radioenlaces
    - 26,453 GHz – 27 GHz: disponible para uso inmediato con limitaciones
    - 27 GHz – 27.5 GHz: disponible para uso inmediato, sin limitaciones

  Sin embargo, el Cuadro Nacional de Atribución de Frecuencias ha establecido como fecha límite el 31 de diciembre de 2021 para que este rango quede liberado

BANDA 3400-3600		
Frecuencias (MHz)		Operadora
3400	3440	Mas movil
3440	3460	Movistar
3460	3480	Orange
3480	3500	Ministerio de Defensa
3500	3540	Mas movil
3540	3560	Movistar
3560	3580	Orange
3580	3590	Movistar
3590	3600	Orange

Tabla 1: Subasta de la Banda 3400 MHz - 3600 MHz

BANDA 3600-3800					
Subasta			Reorganización		
Frecuencias (MHz)		Operadora	Frecuencias (MHz)		Operadora
3600	3610	Vodafone	3600	3660	Orange
3610	3620	Movistar	3660	3750	Vodafone
3620	3640	Orange	3750	3800	Movistar
3640	3650	Vodafone			
3650	3660	Orange			
3660	3670	Vodafone			
3670	3680	Orange			
3680	3720	Movistar			
3720	3780	Vodafone			
3780	3800	Orange			

Tabla 2: Subasta de la Banda 3600 MHz - 3800 MHz

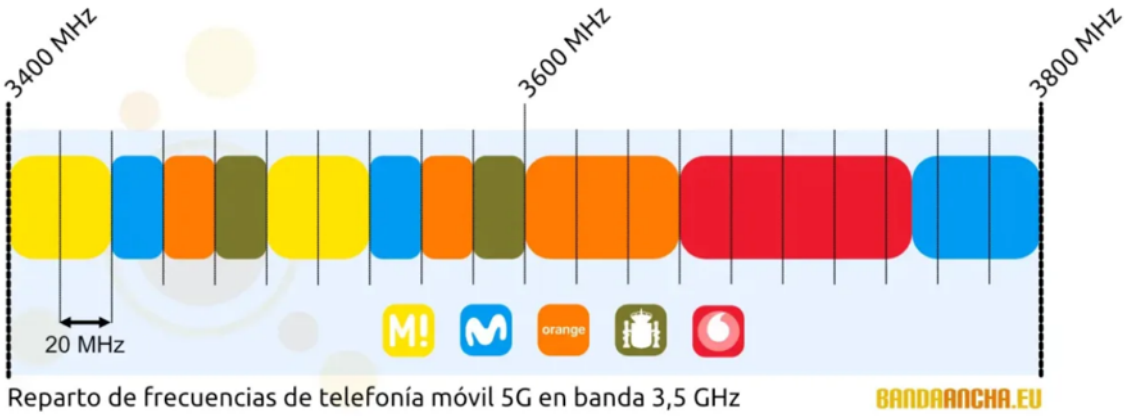


Figura 18: Reparto frecuencias 5G - Banda 3,5 GHz [19]

El Gobierno de España ha determinado como objetivo que en el año 2025 tres cuartos de la población española tengan cobertura 5G. Dependiendo del servicio que se quiera proporcionar, se podrán utilizar unos rangos de frecuencias u otros. Las frecuencias por debajo del GHz se utilizarán para dar una cobertura elevada puesto que existe mayor penetración. Por otra parte, las frecuencias por debajo de los 6 GHz, también pertenecientes al rango de FR1, encontrarán un balance en términos de cobertura y transmisión de datos. Finalmente, frecuencias muy elevadas (mmWave) en el que el radio de cobertura es muy pequeño, podrán utilizarse para realidad virtual, o streaming de alta definición.

3.2. Estado de implantación

Este proyecto también ha querido comprobar el estado de implantación de las redes 5G, tanto NSA como SA. Para poder realizar esta comprobación se utilizo un terminal 5G, en concreto, el Oppo Reno 4z 5G, con su tarjeta SIM correspondiente y se instalaron varias aplicaciones para rastrear los identificadores de las celdas a las que se ha conectado el móvil, su localización, tecnología y velocidad de transmisión. En concreto, se utilizaron GNet-Track, Network Signal Info y LTE Signalling.

A continuación, se realizó un *wardriving* por distintas zonas de la Comunidad de Madrid para obtener la información de las distintas celdas y a que tecnología pertenecían, obteniendo el mapa de calor representado en la Figura 19. Este mapa de calor representa las celdas 5G encontradas de la operadora Movistar de la cual solo se consiguieron trazas de 5G NSA, salvo en el distrito Telefónica ubicado en la M-40 donde se obtuvo 5G-SA.

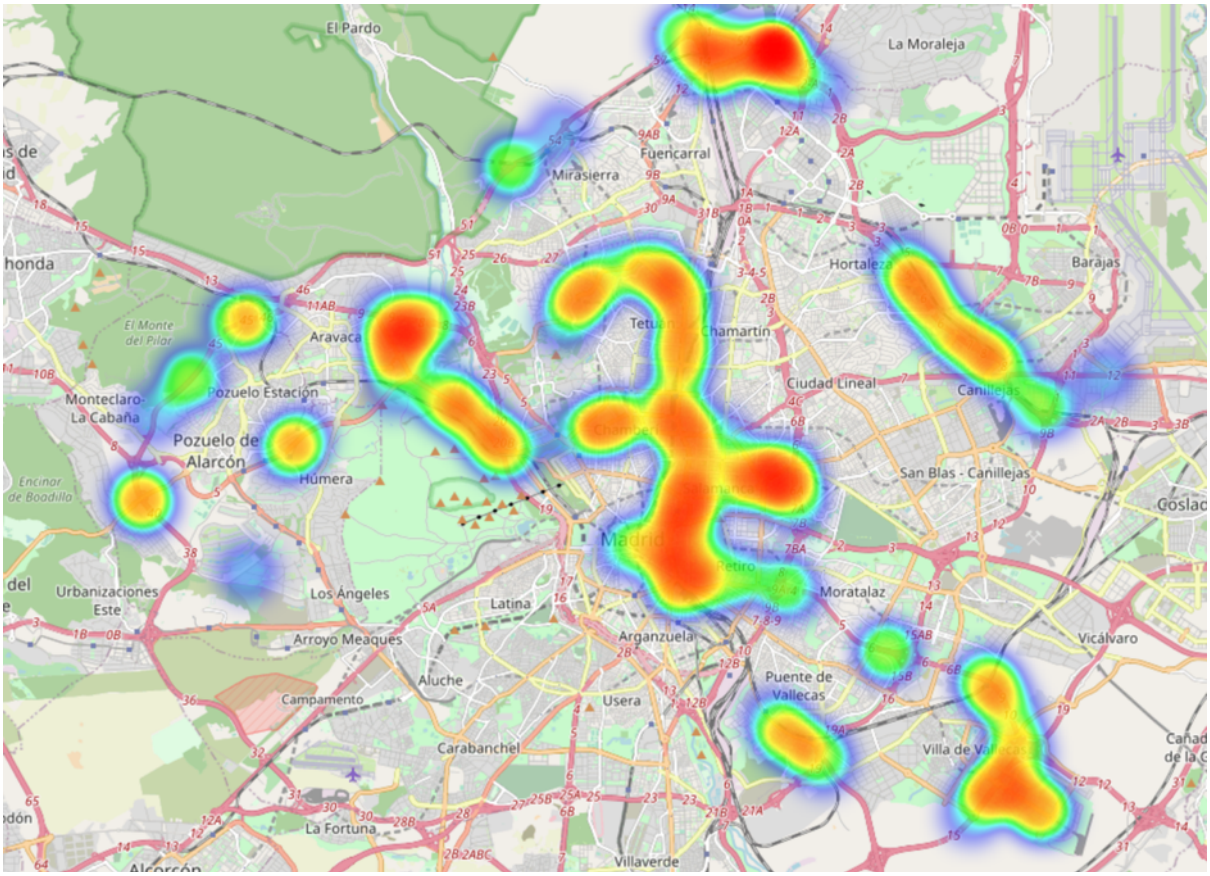


Figura 19: Mapa de calor de redes 5G de Movistar en la Comunidad de Madrid

El wardriving completo se puede inspeccionar en la **Sección 9.5.**

3.3. Soluciones comerciales y Open Source

Desde el punto de vista de la investigación, existen dos opciones principales para abordar las tecnologías móviles, en este caso el 5G. En primer lugar, existe la posibilidad de comprar un SDR, explicado en detalle en la **Sección 4.2**, y utilizar un software libre para recrear en el laboratorio una red móvil completa. Esta posibilidad, dependiendo del SDR adquirido, es en general rentable, pero requiere trabajo de instalación y configuración. La otra posibilidad sería comprar una solución comercial, pay & go, en el que no sería necesario ninguna instalación. Sin embargo, esta solución requiere de un coste mucho más elevado. Es importante destacar que para recrear una red móvil no sirve cualquier SDR, tiene que ser full-duplex, es decir, la capacidad de transmitir y recibir al mismo tiempo. Mucho de los SDRs en el mercado como HackRF o RTL-SDR son half-duplex lo que no serviría. Por tanto, los SDR que se mencionan más adelante son todos full-duplex.

Los principales SDR en el mercado son los siguientes:

- **Ettus USRP**, de los cuales destacamos los siguientes productos:
  - USRP X Series
    - USRP X410 (19 055 €)
    - USRP X310 (6 252 €)
    - USRP X300 (5 078 €)
  - USRP Bus Series (~1000 €)
    - USRP B200/B200 mini
    - USRP B210
  - USRP Networked Series
    - USRP N320 (13 184 €)
    - USRP N300 (7 488 €)
    - USRP N200 (1 978 €)

Figura 20: SDR Ettus Research, de arriba abajo: USRP X300, USRP B210 y USRP N200 [25]

- **Blade-RF** (~1000 €)
- **LimeSDR** (~500 €)

Figura 21: BladeRF [32]

Figura 22: LimeSDR [11]

	Ettus B200	Ettus B210	BladeRF x40	LimeSDR
Frequency Range	70MHz-6GHz	70MHz-6GHz	300MHz-3.8GHz	100kHz-3.8GHz
RF Bandwidth	61.44MHz	61.44MHz	40MHz	61.44MHz
Sample Depth	12 bits	12 bits	12 bits	12 bits
Sample Rate	61.44MSPS	61.44MSPS	40MSPS	61.44MSPS (Limited by USB 3.0 data rate)
Transmitter Channels	1	2	1	2
Receivers	1	2	1	2
Duplex	Full	Full	Full	Full
Interface	USB 3.0	USB 3.0	USB 3.0	USB 3.0
Programmable Logic Gates	75k	100k	40k (115k avail)	40k
Chipset	AD9364	AD9361	LMS6002M	LMS7002M
Open Source	Schematic, Firmware	Schematic, Firmware	Schematic, Firmware	Full
Oscillator Precision	+/-2ppm	+/-2ppm	+/-1ppm	+/-1ppm initial, +/-4ppm stable
Transmit Power	10dBm+	10dBm+	6dBm	0 to 10dBm (depending on frequency)
Price	$686	$1,119	$420 ($650)	$299 ($289 pre-order)

Tabla 3: Comparación SDR Full-duplex [20]

Las soluciones comerciales con más renombre actualmente son las siguientes:

YateBTS

YateBTS es una empresa soluciones en el ámbito de las comunicaciones basándose en Yate (Yet Antoher Telephony Engine) implementando tanto GSM/GPRS y 4G como 5G. Su solución 5G se conoce como *MiniCore 5G*, incluyendo los nodos core de las distintas redes.

Figura 23: YateBTS MiniCore 5G [29]

Amarisoft

Empresa dedicada a las soluciones industriales de telecomunicaciones. Centrada en las soluciones en 4G y 5G proporciona una calidad alta a un precio razonable. Para la tecnología 5G encontramos varios productos:

- Amari Callbox Mini
- Amari Callbox Classic
- Amari Callbox Ultimate

A diferencia de YateBTS la serie Callbox de Amarisoft si incluye el nodo radio, es decir, el eNB y gNB.

Figura 24: Amari Callbox Mini [28]

4. Herramientas

4.1. Open Air Interface (OAI)

Open Air Interface Software Alliance es una organización de software libre centrada en el desarrollo de proyectos de redes de acceso radio (RAN) y redes core (CN). Al ser Open Source, la comunidad de desarrolladores puede colaborar con los proyectos, acelerando su progreso y acercando a la comunidad industrial con la universitaria. OAI comenzó con la implementación de una red LTE, sin embargo, en estos últimos años han decidido dedicar el tiempo al desarrollo de las redes 5G, tanto NSA como SA. La red 5G – NSA está totalmente desarrollada, puesto que se apoya en una infraestructura 4G. Sin embargo, la red 5G – SA todavía está desarrollándose. En este proyecto se va a utilizar el código proporcionado por esta organización para levantar una red 5G y realizar distintas pruebas de seguridad.

4.2. SDR

Los SDR son dispositivos el cual su objetivo es sustituir el hardware por componentes de software. Esto permite una mayor flexibilidad a la hora de programar y procesar señales y ha supuesto una revolución en el desarrollo e investigación de las redes móviles.

4.2.1. Ettus USRP

Ettus Research es una empresa especializada en SDRs e investigación de la tecnología Wireless. En este proyecto se utilizó el Ettus USRP B200 y B200 mini. El USRP B200 tiene una cobertura de frecuencias de 70 MHz a 6 GHz y un ancho de banda de 56 MHz. Además, tiene un precio asequible para la calidad del servicio. Por otra parte, también se utilizo la versión "light" del B200, conocido como B200 mini. Tiene prácticamente las mismas especificaciones que el anterior. La principal diferencia aparte del tamaño es que USRP B200 tiene conector GPS, para estabilizar el reloj de referencia, muy útil en este tipo de experimentaciones. Mucho de los proyectos realizados con Open Air Interface para redes 5G NSA y SA utilizaban el USRP B210, en el que la principal diferencia es que es MIMO (Multiple Input Multiple Output) mientras que los ya mencionados son SISO (Single Input Single Output).

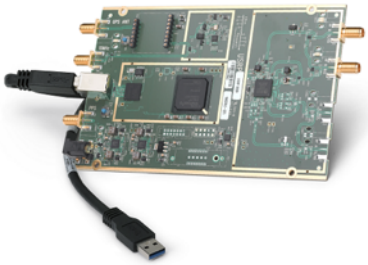


Figura 26: USRP B200 [25]

Figura 25: USRP B200 mini [25]

4.3. Sysmocom

Sysmocom (Systems for Mobile Communications GmbH) es una empresa que desarrolla productos y servicios de telecomunicaciones y software libre.

En el caso de este proyecto, se utilizaron las tarjetas SIM – sysmoISIM-SJA2. Estas tarjetas son la evolución de las sysmoUSIM-SJS1, tarjetas utilizadas para las tecnologías 2G, 3G y 4G; en las cuales se han añadido y mejorado los siguientes aspectos relevantes a este proyecto:

- Adición de la aplicación ISIM para IMS/VoLTE a parte de SIM + USIM
- Parámetros relacionados con las redes 5G, permitiendo el acceso a dichas redes

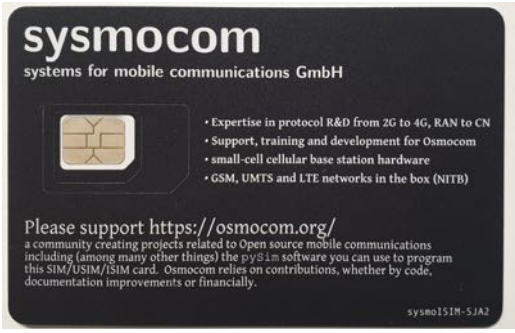


Figura 27: sysmoisim-sja2 tarjeta SIM [21]

Estas tarjetas SIM tienen la capacidad de ser reprogramadas mediante un programa open source desarrollado por sysmocom, especificado en su manual [22], en la **Sección 7.2** y **7.3**. Para permitir a estas tarjetas SIM funcionar con Open Air Interface, era necesario realizar unos cambios, por lo que había que reprogramarlas. Para poder reprogramar las tarjetas SIM es necesario conocer su

clave ADM1, que es la clave administrativa. Sin ella, será imposible modificar otros parámetros de la tarjeta.

Para realizar los siguientes cambios es necesario descargarse el repositorio, especificado en la **Sección 9.3.5**.

En primer lugar, el software de OAI utiliza el algoritmo de autenticación MILENAGE, por lo que había que especificar a la tarjeta SIM que la aplicación USIM usase ese mismo algoritmo. Existían varias posibilidades de algoritmos:

- COMP128v1
- COMP128v2
- XOR-2G
- MILENAGE
- SHA1-AKA
- XOR

Por defecto, las tarjetas venían con el algoritmo COMP128v1:

```
(venvpython2) miguel@miguel-NUC10i7FNH:~/Documents/OAI/progrom_sim_cards/sysmo-usim-tool$ ./sysmo-isim-tool.sja2.py -a 34842566 -t

sysmoISIM-SJA2 parameterization tool
Copyright (c)2019 Sysmocom s.f.m.c. GmbH

Trying to find card with ATR: 3B 9F 96 80 1F 87 80 31 E0 73 FE 21 1B 67 4A
4C 75 30 34 05 4B A9
Initializing smartcard terminal...
 * Detected Card IMSI:  901700000049607
   ISIM Application installed
   USIM Application installed

Authenticating...
 * Remaining attempts: 3
 * Authenticating...
 * Authentication successful
 * Remaining attempts: 3

Reading Authentication parameters...
 * Initalizing...
 * Reading...
 * Current algorithm setting:
   2G: 1=COMP128v1
   3G: 1=COMP128v1

Done!
```

Por tanto, había que modificarlo a MILENAGE:

```
(venvpython2) miguel@miguel-NUC10i7FNH:~/Documents/OAI/progrom_sim_cards/sysmo-usim-tool$ ./sysmo-isim-tool.sja2.py -a 34842566 -T MILENAGE:MILENAGE

sysmoISIM-SJA2 parameterization tool
Copyright (c)2019 Sysmocom s.f.m.c. GmbH
```

```
Trying to find card with ATR: 3B 9F 96 80 1F 87 80 31 E0 73 FE 21 1B 67 4A
4C 75 30 34 05 4B A9
Initializing smartcard terminal...
 * Detected Card IMSI:  901700000049607
   ISIM Application installed
   USIM Application installed

Authenticating...
 * Remaining attempts: 3
 * Authenticating...
 * Authentication successful
 * Remaining attempts: 3

Programming Authentication parameters...
 * Initalizing...
 * New algorithm setting:
   2G: 4=MILENAGE
   3G: 4=MILENAGE
 * Programming...

Done!
```

Por otra parte, había que modificar el uso de una clave de operador conocido como Operator Code - OP. Al realizar el pedido de tarjetas SIM, sysmocom proporciona las claves e identidad de estas para poder reprogramarlas. Este número se utiliza para la generación de claves. Este valor, de 128 bits, se utiliza en el algoritmo de encriptación para generar el OPc, el cual esta cifrado, y es el que verdaderamente se utiliza para el posterior cifrado. El OPc se calcula con la siguiente fórmula,

$$AES128\ (K_i\ , OP)\ XOR\ OP = OP_c$$

Donde Ki es una de las claves de la tarjeta SIM y AES128 es un estándar de cifrado con una clave de 128 bits.

En el caso de este proyecto el proveedor de tarjetas SIM solo proporcionaba el OPc cuando en el software de OAI, al configurar el nodo HSS, había que especificar el OP. Dentro de la tarjeta SIM se especifica un booleano para saber que valor utilizar, si el OP o el OPc. Las tarjetas provisionadas por sysmocom tenían por defecto este valor para utilizar el OPc. Por ello, era necesario programar la tarjeta para que la tarjeta SIM utilizase el OP.

```
(venvpython2) miguel@miguel-NUC10i7FNH:~/Documents/OAI/progrom_sim_cards/sysmo-
usim-tool$ ./sysmo-isim-tool.sja2.py -a 34842566 -o

sysmoISIM-SJA2 parameterization tool
Copyright (c)2019 Sysmocom s.f.m.c. GmbH

Trying to find card with ATR: 3B 9F 96 80 1F 87 80 31 E0 73 FE 21 1B 67 4A
4C 75 30 34 05 4B A9
Initializing smartcard terminal...
 * Detected Card IMSI:  901700000049607
   ISIM Application installed
   USIM Application installed
```

```
Authenticating...
 * Remaining attempts: 3
 * Authenticating...
 * Authentication successful
 * Remaining attempts: 3

Reading OP/c value...
 * Initalizing...
 * Reading...
 * Current OP/OPc setting:
   OPc: 70b44fb5994138fd9ebadb1d486f6f9c

Done!
```

```
(venvpython2) miguel@miguel-NUC10i7FNH:~/Documents/OAI/progrom_sim_cards/sysmo-
usim-tool$ ./sysmo-isim-tool.sja2.py -a 34842566 -O
70b44fb5994138fd9ebadb1d486f6f9c

sysmoISIM-SJA2 parameterization tool
Copyright (c)2019 Sysmocom s.f.m.c. GmbH

Trying to find card with ATR: 3B 9F 96 80 1F 87 80 31 E0 73 FE 21 1B 67 4A
4C 75 30 34 05 4B A9
Initializing smartcard terminal...
 * Detected Card IMSI:  901700000049607
   ISIM Application installed
   USIM Application installed

Authenticating...
 * Remaining attempts: 3
 * Authenticating...
 * Authentication successful
 * Remaining attempts: 3

Writing OP value...
 * Initalizing...
 * New OPc setting:
   OP: 70b44fb5994138fd9ebadb1d486f6f9c
 * Programming...

Done!
```

4.4. Wireshark

Wireshark es un programa gratuito para analizar trazas de red en tiempo real. En este proyecto ha sido fundamental para poder observar y examinar en detalle tanto los distintos protocolos utilizados como los mensajes intercambiados entre los nodos. Este software intercepta el tráfico de red y lo convierte a un formato entendible.

5. Ataques identidad abonados 5G

5.1. IMSI Catcher

Como se ha mencionado anteriormente, el número IMSI es información crítica de un usuario. Un IMSI Catcher es un ataque para interceptar este número durante el procedimiento de acople a la red y así poder rastrear la localización del usuario, o incluso interceptar mensajes de texto, llamadas o trafico. Por tanto, la obtención del IMSI es considerada una vulnerabilidad crítica.

La idea detrás de este ataque es simular una red móvil o estación base y hacer que el abonado se intente conectar a esta red. No es necesario ni que se conecte puesto que solo se necesitará el IMSI para autenticar al usuario y este no estará registrado en la base de datos por lo que será rechazado. Estos dispositivos aparecieron por primera vez en la década de los 90 pero eran dispositivos grandes, pesados y caros. Sin embargo, a lo largo de los años, con la aparición de software libre y los SDR, hoy en día se pueden realizar IMSI Catchers por un valor muy comedido.

Existen diversas formas para que un usuario se intente conectar a la estación base falsa. En primer lugar, propagar una señal radio con más potencia y así el UE intentará conectarse, siempre y cuando los datos de la operadora coincidan con la tarjeta SIM. Otra forma sería inhibiendo las frecuencias de la tecnología a la que se quiere atacar y así obligar al UE a escanear todas las frecuencias de nuevo.

Los IMSI Catcher llevan existiendo desde la tecnología 2G, donde la seguridad es escasa y los dispositivos móviles se conectan a "cualquier" red. Sus sucesoras, es decir, tanto 3G como 4G, implementar los IMSIs temporales como se ha descrito en la **Sección 2.4** pero en varias ocasiones el IMSI tiene que ser enviado en texto claro. Por tanto, los IMSI Catchers también funcionan en estas tecnologías.

5.2. Solución propuesta

5.2.1. Caja de Faraday

La norma establece unos requisitos para poder experimentar en el laboratorio con SDRs y tecnologías móviles, entre ellos destaca el uso de una caja de Faraday. Esto se debe a que las frecuencias utilizadas podrían interferir con otros usuarios cercanos a los dispositivos y se podrían ver afectados. La caja de Faraday actúa como un entorno en el que no pueden salir ni entrar radiofrecuencias por lo que se tendría una red completamente aislada.

Figura 28: Caja de Faraday utilizada

5.2.2. Requerimientos de hardware y software

OAI exigía unos mínimos para poder instalar su software y ejecutarlo sin problemas. Para el hardware requería de hosts con procesadores Intel para funciones DSP que hacen un uso de instrucciones SIMD. Los softwares compatibles eran:

- Generaciones 3/4/5/6 Intel Core i5,i7
- Generaciones 2/3/4 Intel Xeon
- Intel Atom Rangeley, E38xx, x5-z8300

Además, se requerían como mínimo 4 núcleos y una versión mínima de Ubuntu de 16.04 para las ramas de git utilizadas.

Los SDR compatibles:

- USRP B210
- USRP X310
- BladeRF
- LimeSDR
- EURECOM EXPRESSMIMO2 RF

Los UE comerciales que se habían testeado eran los siguientes:

- Oppo Reno 5G
- Samsung A90 5G
- Samsung A42 5G
- Google Pixel 5G (note1)
- Simcom SIMCOM8200EA
- Quectel RM500Q-GL

5.2.3. NSA

5.2.3.1. *Arquitectura del sistema*

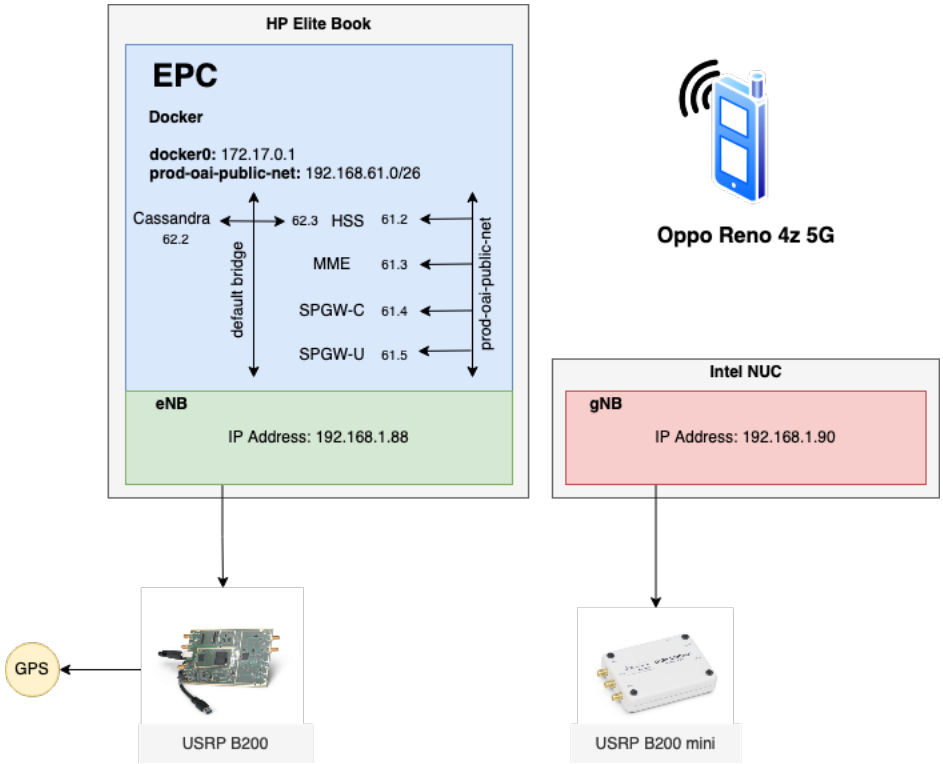


Figura 29: Arquitectura 5G-NSA

El escenario de experimentación fue el siguiente:

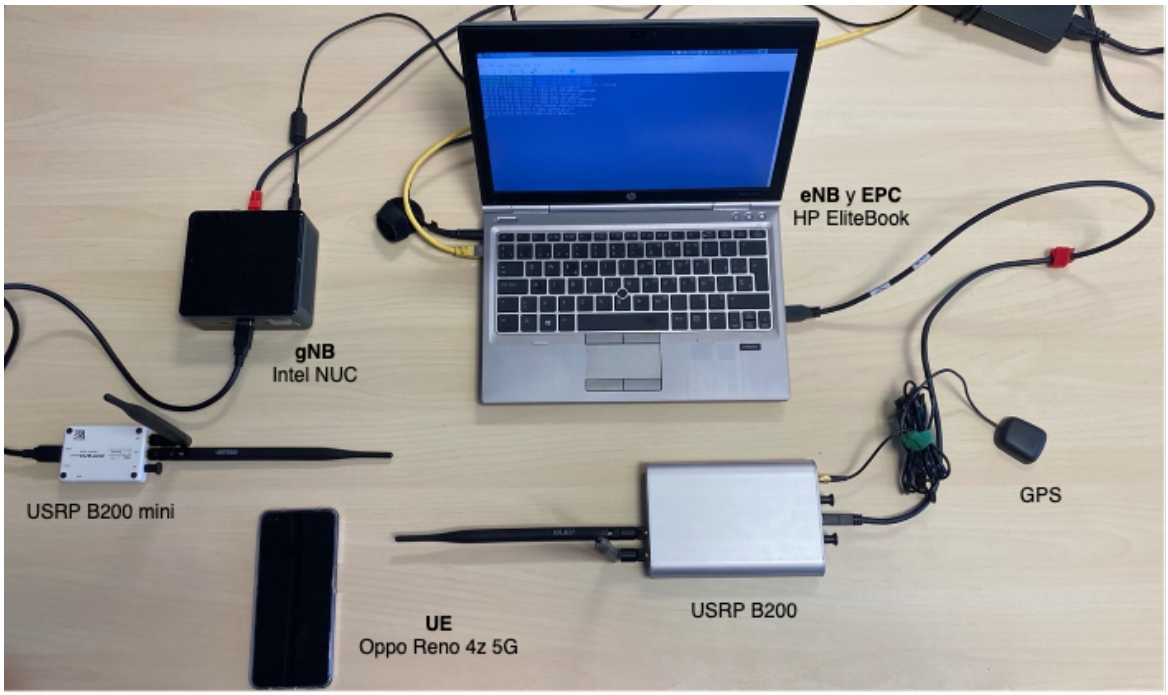


Figura 30: Escenario de laboratorio

En primer lugar, antes de instalar el software era necesario varios hosts con ciertos requerimientos. Como se puede observar en la Figura 29, en este proyecto se necesitan dos hosts diferentes para albergar distintos, uno donde se instalar el EPC y el eNB y otro para instalar el gNB.

Se decidió tener esta arquitectura para ahorrar recursos, puesto que en la documentación provista por OAI se utilizaban tres hosts, uno para cada nodo. En la siguiente tabla se detallan las especificaciones de los ordenadores utilizados para la experimentación.

Nodo	eNB y docker	gNB
Modelo	HP EliteBook 2570p	Intel NUC
CPU	Intel® Core™ i5-3340M CPU @ 2.70 GHz	Intel® Core™ i7-10710U CPU @ 1.10GHz
Memory	256 GB	120 GB
OS	Ubuntu 18.04 LT 64 bits	Ubuntu 18.04 LT 64 bits
Kernel	5.4.0-77-generic	4.15.0-147-generic
Version USB	3.0	3.0

Tabla 4: Especificaciones de hardware utilizado

Por otra parte, en cuanto a los SDR existían varias alternativas, pero finalmente se opto por un USRP B200 y un USRP B200 mini como se puede observar en la Figura 29.

Como UE se eligió el **Oppo Reno 4z 5G**, que tenía las siguientes especificaciones técnicas del modem [23]:

- **2G GSM** 850 / 900 / 1800 / 1900
- **GPRS**
- **EDGE**
- **3G HSDPA** 800 / 850 / 900 / 1700(AWS) / 1900 / 2100
- **4G LTE** B1 / B2 / B3 / B4 / B5 / B7 / B8 / B12 / B17 / B18 / B19 / B20 / B26 / B28 / B32 / B38 / B39 / B40 / B41 / B42 / B66
- **5G SA/NSA** B1 / B3 / B5 / B7 / B8 / B20 / B28 / B38 / B40 / B41 / B77 / B78

5.2.3.2. Ejecución del IMSI Catcher

Para la simulación de una red de laboratorio 5G NSA se siguieron los siguientes pasos, descritos en detalle en la **Sección 9.3**:

1. En primer lugar, se instaló docker, necesario para la configuración del core de la red (EPC).
2. Se descargo el código fuente de los repositorios correspondientes. En este caso eran necesarios dos repositorios: uno para el EPC y otro para los nodos eNB y gNB.
3. Para la configuración del EPC era necesario en primer lugar descargar las imágenes de los siguientes nodos

3.1. Cassandra
3.2. HSS
3.3. MME
3.4. SPGWC
3.5. SPGWU

4. Se configuraron estas imágenes
5. Se configuraron los nodos eNB y gNB para que se pudiesen comunicar entre ellos

Antes de continuar con los siguientes pasos es importante comprobar la interconexión entre los distintos nodos. Para que puedan hablar entre ellos es crucial ejecutar los siguientes comandos.

En el nodo donde se alberga el EPC:

```
$ sudo sysctl net.ipv4.conf.all.forwarding=1
$ sudo iptables -P FORWARD ACCEPT
```

En los nodos eNB y gNB

```
$ sudo ip route add 192.168.61.0/24 via 192.168.1.88 dev wlo1
```

Donde:

- 192.168.61.0/24 es la dirección IP de la red virtual creada en el host de Docker
- 192.168.1.88 es la dirección IP del Docker host
- wlo1 es la interfaz de red del servidor eNB/gNB

Después de ejecutar estos comandos se puede comprobar la interconexión mediante el comando en Linux *ping*

```
ping -c 5 192.168.61.3
PING 192.168.1.1 (192.168.1.1): 56 data bytes
64 bytes from 192.168.1.1: icmp_seq=0 ttl=64 time=1.633 ms
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.499 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=58.499 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=3.169 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=166.768 ms

--- 192.168.1.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.499/46.314/166.768/64.069 ms
```

6. Se conectaron los SDR en los hosts correspondientes. Es necesario conectarlos a un USB 3.0
7. Ejecución del core de la red a través de docker

```
$ sudo docker run --name prod-cassandra -d -e CASSANDRA_CLUSTER_NAME="OAI
HSS Cluster" \
             -e CASSANDRA_ENDPOINT_SNITCH=GossipingPropertyFileSnitch
cassandra:2.1
```

```
$ sudo docker run --privileged --name prod-oai-hss -d --entrypoint
/bin/bash oai-hss:production -c "sleep infinity"
$ sudo docker network connect prod-oai-public-net prod-oai-hss
$ sudo docker run --privileged --name prod-oai-mme --network prod-oai-
public-net \
             -d --entrypoint /bin/bash oai-mme:production -c "sleep
infinity"
$ sudo docker run --privileged --name prod-oai-spgwc --network prod-oai-
public-net \
             -d --entrypoint /bin/bash oai-spgwc:production -c "sleep
infinity"
$ sudo docker run --privileged --name prod-oai-spgwu-tiny --network prod-
oai-public-net \
             -d --entrypoint /bin/bash oai-spgwu-tiny:production -c "sleep
infinity"
```

8. Ejecución del eNB

```
~/openairinterface5g/cmake_targets/ran_build/build$ sudo ./lte-softmodem -O
**YOUR_ENB_CONF_FILE**
```

9. Ejecución del gNB

La opción -E es necesaria si se utiliza un USRP B2xxx, no está soportada cuando se utilice un N300 USRP

```
~/openairinterface5g/cmake_targets/ran_build/build$ sudo ./nr-softmodem -O
**YOUR_GNB_CONF_FILE** -E
```

Una vez ejecutado, los nodos radio se comunicarán entre ellos con un primer mensaje en las que el gNB se da a conocer.

Cuando un UE se intente conectar al eNB, se recibirá el siguiente mensaje:

```
[RRC]   [FRAME 00218][eNB][MOD 00][RNTI c63a] Accept new connection from UE
random UE identity (0xf106d192d1000000) MME code 0 TMSI 0 cause 3
```

Este mensaje corresponde con el RRC Connection Request. Como se puede comprobar, puesto que es la primera vez que el UE se intenta conectar a esta red, enviará una identidad aleatoria – *random UE identity* -. Una vez se comprueba si ha llegado este mensaje de intento de conexión, se verifica en los logs del MME el IMSI que se habrá enviado en el Attach Request.

Exista o no exista en la base de datos, puesto que es la primera vez que se intente conectar a esta red, se tendrá que enviar el IMSI en texto claro.

```
000536 00049:345289 7F5F188FA700 DEBUG NAS-EM r-
mme/src/nas/emm/emm_data_ctx.c:0197    ue_id=1 set IMSI 901700000013638
(valid)
```

Con este mensaje, ya obtenemos el IMSI del usuario y podríamos perpetrar numerosos ataques adicionales.

El proyecto se realizó en BASH debido a su facilidad para ejecutar comandos del sistema operativo y coordinar distintos scripts. Se decidió automatizar todo el proceso de arranque y parada de los distintos nodos, tanto eNB, gNB como el core de la red. Mientras, un watchdog iba comprobando si algún dispositivo se había intentado conectar a la red. En caso afirmativo, comprobaba las trazas en el EPC para buscar y determinar el IMSI del cliente.

Se decidió general un archivo de LOG para poder revisar el IMSI Catcher y ver que dispositivos se habían intentado conectar.

```
2021-07-20 13:19:04 start_nsa.sh [INFO] Received command to start 5G - NSA
(Non-StandAlone)
2021-07-20 13:19:04 configure_nsa.sh [INFO] Enabled IP Routing
2021-07-20 13:19:04 configure_nsa.sh [INFO] eNB USRP detected!
2021-07-20 13:19:08 configure_nsa.sh [INFO] eNB USRP probe succesfull!
2021-07-20 13:19:08 configure_nsa.sh [INFO] eNB configuration OK
2021-07-20 13:19:12 configure_nsa.sh [INFO] gNB is REACHABLE
2021-07-20 13:19:15 configure_nsa.sh [INFO] gNB configuration OK
2021-07-20 13:19:47 configure_nsa.sh [INFO] Docker configuration OK
2021-07-20 13:19:47 start_nodes.sh [INFO] Received command to start eNB and
gNB
2021-07-20 13:19:47 start_nodes.sh [INFO] eNB USRP detected!
2021-07-20 13:19:51 start_nodes.sh [INFO] eNB USRP probe succesfull!
2021-07-20 13:19:13 configure_gnb.sh [INFO] USRP detected!
2021-07-20 13:19:15 configure_gnb.sh [INFO] USRP probe succesfull!
2021-07-20 13:19:54 start_gnb.sh [INFO] USRP detected!
2021-07-20 13:19:56 start_gnb.sh [INFO] USRP probe succesfull!
2021-07-20 13:19:56 start_nodes.sh [INFO] gNB started OK
2021-07-20 13:20:01 imsi_catcher.sh [INFO] DEBUG: ENB_LINES=538
2021-07-20 13:20:01 imsi_catcher.sh [INFO] THERE HAVE BEEN 0 CONNECTIONS
2021-07-20 13:20:01 imsi_catcher.sh [INFO] No UE connections yet...
2021-07-20 13:21:02 imsi_catcher.sh [INFO] DEBUG: ENB_LINES=16
2021-07-20 13:21:02 imsi_catcher.sh [INFO] THERE HAVE BEEN 0 CONNECTIONS
2021-07-20 13:21:02 imsi_catcher.sh [INFO] No UE connections yet...
2021-07-20 13:22:02 imsi_catcher.sh [INFO] DEBUG: ENB_LINES=10
2021-07-20 13:22:02 imsi_catcher.sh [INFO] THERE HAVE BEEN 0 CONNECTIONS
2021-07-20 13:22:02 imsi_catcher.sh [INFO] No UE connections yet...
[0;33m 2021-07-20 13:23:01 watchdog_nsa.sh [WARNING] Node eNB was NOT
stable, restarting nodes... [0m
2021-07-20 13:23:01 watchdog_nsa.sh [INFO] eNB was stable during - 00:03:10
2021-07-20 13:23:01 stop_nodes.sh [INFO] Stopping both nodes eNb and gNB
2021-07-20 13:23:03 stop_nodes.sh [INFO] eNB stopped OK
2021-07-20 13:19:13 configure_gnb.sh [INFO] USRP detected!
2021-07-20 13:19:15 configure_gnb.sh [INFO] USRP probe succesfull!
2021-07-20 13:19:54 start_gnb.sh [INFO] USRP detected!
2021-07-20 13:19:56 start_gnb.sh [INFO] USRP probe succesfull!
2021-07-20 13:23:14 start_nodes.sh [INFO] Received command to start eNB and
gNB
2021-07-20 13:23:14 start_nodes.sh [INFO] eNB USRP detected!
2021-07-20 13:23:18 start_nodes.sh [INFO] eNB USRP probe succesfull!
2021-07-20 13:19:13 configure_gnb.sh [INFO] USRP detected!
2021-07-20 13:19:15 configure_gnb.sh [INFO] USRP probe succesfull!
2021-07-20 13:19:54 start_gnb.sh [INFO] USRP detected!
2021-07-20 13:19:56 start_gnb.sh [INFO] USRP probe succesfull!
2021-07-20 13:23:21 start_gnb.sh [INFO] USRP detected!
2021-07-20 13:23:23 start_gnb.sh [INFO] USRP probe succesfull!
2021-07-20 13:23:23 start_nodes.sh [INFO] gNB started OK
```

```
2021-07-20 13:24:02 imsi_catcher.sh [INFO] DEBUG: ENB_LINES=598
2021-07-20 13:24:02 imsi_catcher.sh [INFO] THERE HAVE BEEN 1 CONNECTIONS
2021-07-20 13:24:02 imsi_catcher.sh [INFO] DEBUG: FIND_IMSI: set IMSI
214075551043473
2021-07-20 13:25:01 imsi_catcher.sh [INFO] DEBUG: ENB_LINES=12
2021-07-20 13:25:01 imsi_catcher.sh [INFO] THERE HAVE BEEN 0 CONNECTIONS
2021-07-20 13:25:01 imsi_catcher.sh [INFO] No UE connections yet...
2021-07-20 13:26:02 imsi_catcher.sh [INFO] DEBUG: ENB_LINES=12
2021-07-20 13:26:02 imsi_catcher.sh [INFO] THERE HAVE BEEN 0 CONNECTIONS
2021-07-20 13:26:02 imsi_catcher.sh [INFO] No UE connections yet...
2021-07-20 13:26:13 stop_nsa.sh [INFO] Received command to stop NSA 5G
2021-07-20 13:26:13 stop_nodes.sh [INFO] Stopping enb node
2021-07-20 13:26:15 stop_nodes.sh [INFO] eNB stopped OK
```

Por otra parte, se iban almacenando los IMSIS descubiertos a lo largo de la experimentación:

```
2021-07-20 12:44:02 -> Found IMSI: 214075551043473, MNC: 214, MCC: 07
```

5.2.4. SA

Para la opción 5G-SA era necesario descargarse una nueva versión del nodo gNB y desplegar mediante docker el nuevo core de la red.

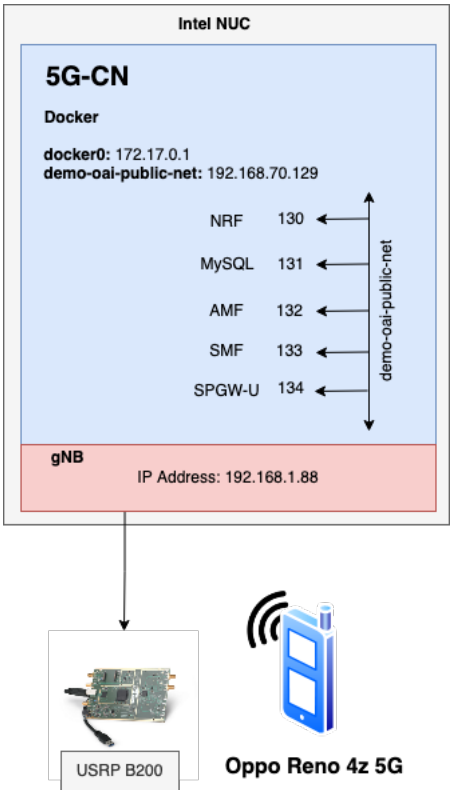


Figura 31: Arquitectura 5G-SA

Para la ejecución del despliegue SA, se siguieron los siguientes pasos, descritos en detalle en la **Sección 9.3.3.**

1. En primer lugar, se instalo docker, necesario para la ejecución de los nodos core de la red
2. Se descargó el código fuente del nodo gNB
3. Se descargarón las imágenes de los nodos principales:
    a. AMF
    b. SMF
    c. NRF
    d. SPGWU
4. A diferencia del EPC, se configuraron las imágenes mediante el archivo *docker-compose.yaml*
5. Una vez instalados todos los componentes, se ejecutaron en el orden correspondiente.
    a. En primer lugar, el core de la red:

```
$ sudo oai-cn5g-fed/docker-compose/core-network.sh start nrf

Starting 5gcn components in the order nrf, mysql, amf, smf, spgwu...
Creating oai-nrf ... done
Creating mysql     ... done
Creating oai-spgwu ... done
Creating oai-smf   ... done
Creating oai-amf    ... done
Creating oai-ext-dn ... done
Core network started, checking the health status of the containers...
Checking the if the containers are configured...

Checking if SMF and UPF registered with nrf core network

For example: oai-smf Registration with oai-nrf can be checked on this url
/nnrf-nfm/v1/nf-instances?nf-type='SMF' {"_links":{"item":[],"self":""}}
SMF and UPF are registered to NRF...

Core network is configured and healthy, total time taken 6391 milli seconds
```

    b. En segundo lugar, el gNB. Como se ha mencionado en la arquitectura NSA, puesto que se utilizaba un USRP era necesario ejecutar el binario con la opción **-E** además de la opción **--sa** por estar en este tipo de despliegue.

```
$ cd cmake_targets/ran_build/build
$ sudo ./nr-softmodem -E --sa -O <config_file>
```

Para la ejecución correcta de los nodos, es importante que tanto el archivo de configuración del gNB como el amf tengan los mismos valores de MCC, MNC y TAC. En una primera instancia, el gNB se comunica con el AMF mediante los siguientes mensajes:

No.	Time	Delta	Source	Destination	Protocol	Length	Info
6136	21.094633288	0.000000000	192.168.70.129	192.168.70.132	NGAP	138	NGSetupRequest
6138	21.095831040	0.001197752	192.168.70.132	192.168.70.129	NGAP	574	NGSetupResponse

Sin embargo, la única comunicación que existía entre el UE y el gNB era el Msg1, es decir, el Random Preamble que envía el UE.

```
[0m[0m[PHY]   [gNB 0][RAPROC] Frame 175, slot 19 Initiating RA procedure
with preamble 33, energy 31.7 dB, delay 23 start symbol 0 freq index 0
[0m[0m[MAC]   UL_info[Frame 176, Slot 0] Calling initiate_ra_proc
RACH:SFN/SLOT:175/19
[0m[0m[1;31m[MAC]   [gNB 0][RAPROC] FAILURE: CC_id 0 Frame 175 initiating
RA procedure for preamble index 33
[0m[0m[PHY]   [gNB 0][RAPROC] Frame 199, slot 19 Initiating RA procedure
with preamble 53, energy 36.7 dB, delay 28 start symbol 0 freq index 0
```

Por tanto, no era posible comprobar el Registration Request con la versión actual de Open Air Interface.

6. Análisis de los resultados

En la arquitectura 5G-NSA se podían realizar IMSI-Catcher siempre y cuando el móvil tuviese capacidad para usar las bandas 5G adecuadas. Cuando un dispositivo móvil se conecta a una red 5G-NSA utilizará el procedimiento de LTE, donde se envía el IMSI en texto claro si es la primera vez que se conecta a dicha. Esto es una vulnerabilidad crítica, puesto que mediante el IMSI se podría localizar al usuario. El IMSI Catcher funcionaba correctamente pero no totalmente de forma pasiva, puesto que había que desactivar manualmente el modo avión del dispositivo.

A lo largo de la experimentación de la red de laboratorio se dieron diversos casos en la que esta fallaba. Como se ha descrito anteriormente, en la arquitectura NSA se utilizaban dos hosts, uno que albergara el core de la red con docker y el eNB y otro que almacenase el gNB. El nodo que daba problemas de vez en cuando era el eNB porque el ordenador HP no proporcionaba suficiente transmisión de datos para que fuese estable. En la documentación de OAI se especificaba como recomendable utilizar un host para cada nodo de red, es decir, tres diferentes. Por tanto, como recomendación para usos futuros, está será la arquitectura a seguir. Un ejemplo error que salía por pantalla era el siguiente:

```
[PHY] L1_thread isn't ready in 1.1, aborting RX processing
[PHY] L1_thread isn't ready in 1.2, aborting RX processing
[PHY] L1_thread isn't ready in 1.3, aborting RX processing
[PHY] L1_thread isn't ready in 1.4, aborting RX processing
[PHY] L1_thread isn't ready in 1.5, aborting RX processing
[PHY] L1_thread isn't ready in 1.6, aborting RX processing
[PHY] L1_thread isn't ready in 1.7, aborting RX processing
                                        .
                                        .
                                        .
```

En cuanto al estado de implantación, la red de Movistar desplegada en la Comunidad de Madrid es prácticamente en su totalidad 5G-NSA. Para hacer el wardriving, se utilizo el Oppo Reno con la aplicación G-Net Track Pro. Las especificaciones señalan que muchas veces los valores desplegados en la aplicación como el número de celda a la que se ha conectado el móvil o la frecuencia que está utilizando varían dependiendo del dispositivo. Por tanto, para haber obtenido un mapa de calor más preciso hubiera sido óptimo utilizar diferentes dispositivos móviles de quinta generación y distintas aplicaciones que permitiesen analizar los valores de celdas. Además, hubiese sido interesante efectuar este análisis con todas las operadoras principales del país y asi comparar el despliegue de esta tecnología en su totalidad. Sin embargo, los resultados obtenidos demuestran que todavía la implantación de las redes 5G están en una fase prematura y falta mucho desarrollo.

7. Conclusiones

Se ha podido comprobar como en el despliegue 5G-NSA si que es posible perpetrar un IMSI Catcher; en concreto, en el despliegue donde el eNB es el nodo maestro y ambos nodos radio están conectados al core 4G. Esto es debido a que UE se conecta con el mismo procedimiento que en la tecnología del nodo eNB, es decir, 4G-LTE. Aunque en esta tecnología si que es cierto que se utilizan identificadores temporales como el GUTI, si un usuario se conecta por primera vez a una red, tiene que identificarse con el IMSI, enviándolo en texto claro. Esto hace que la interceptación de este número sea posible, vulnerando la privacidad de los usuarios.

Sin embargo, en 5G-SA no se ha podido comprobar puesto que el desarrollo por parte de OAI todavía es muy prematuro y no se alcanzan los mensajes necesarios, como el Identity Request/Response en los que se intercambia la información crítica del usuario. Por tanto, habrá que esperar hasta que alguna organización como Open Air Interface terminen de implementar el conjunto del core de la red 5G y el nodo radio gNB.

Por otra parte, el estado de implantación de las redes 5G todavía está en una etapa muy inicial. Si existen despliegues 5G, son del tipo NSA, es decir, apoyándose en la infraestructura 4G. Todavía queda mucha inversión y desarrollo para que la nueva infraestructura sea implantada por las operadoras, por tanto, todavía es posible atacar a los usuarios mediante un IMSI Catcher además de que las especificaciones todavía están en proceso de desarrollo.

Otra reflexión referente al 5G es si las operadoras desarrollarán su infraestructura teniendo en cuenta la virtualización. Aunque es cierto que esto reduce los costes de forma considerable, tanto SDN y NFV son tecnologías prematuras y complicadas de desarrollar. Además, no existen unos estándares que las operadoras puedan seguir para implantar su infraestructura debido a las infinitas posibilidades en el despliegue. Por tanto, queda en el aire la implantación de estas tecnologías de virtualización.

7.1. Futuros estudios

Los futuros estudios se centrarán en comprobar si las especificaciones descritas por 3GPP se van a cumplir a nivel práctico o no, es decir, si el IMSI Catcher se podrá realizar en el despliegue 5G – SA. Esto se podrá realizar de varias formas. Esperar a que OAI u otro software libre como srsRAN desarrolle 5G-SA o utilizar una solución comercial como la serie Callbox de Amarisoft, puesto que incluye tanto la red core como los nodos radio. Habrá que comprobar que Release de 3GPP se ha utilizado en el producto y ver si existen algunos cambios respecto a la seguridad en las publicaciones posteriores.

8. Bibliografía

[1] J. Pico Garcia y D. Perez Conde, Hacking y seguridad en comunicaciones móviles GSM/GPRS/UMTS/LTE, Informática64, 2011.

[2] «The Network Access Identifier,» [En línea]. Available: https://datatracker.ietf.org/doc/html/rfc4282.

[3] «5G Identifiers SUPI and SUCI,» 2019. [En línea]. Available: https://www.techplayon.com/5g-identifiers-supi-and-suci/.

[4] ShareTechnote, «5G/NR - RAN Architecture,» [En línea]. Available: https://www.sharetechnote.com/html/5G/5G_RAN_Architecture.html.

[5] O. S. Larry Peterson, 5G Mobile Networks: A Systems Approach.

[6] P. H. M. O. L. F. S. S. C. M. Stefan Rommer, 5G Core Networks Powering Digitalization, ELSEVIER, 2020.

[7] SAMSUNG, «5G Standalone Architecture,» 2021.

[8] «5G System Overview,» APIS - apistraining.com.

[9] vmware, «Redes definidas por software,» vmware, [En línea]. Available: https://www.vmware.com/es/topics/glossary/content/software-defined-networking.html.

[10] O. N. FOUNDATION, «Software-Defined Networking: The New Norm for Networks,» 2012.

[11] «Network Functions Virtualization Overview - vmware Docs,» 2019. [En línea]. Available: https://docs.vmware.com/en/VMware-vCloud-NFV/2.0/vmware-vcloud-nfv-openstack-edition-ra20/GUID-FBEA6C6B-54D8-4A37-87B1-D825F9E0DBC7.html.

[12] S. Sirotkin, «5G Radio Access Network Architecture - The Dark Side of 5G,» 2021 John Wiley & Sons Ltd. , 2021.

[13] P. D. L. J. R.-M. J. L. J. F. J. Ordonez-Lucena, «Network Slicing for 5G with SDN/NFV: Concepts, Architecturesand Challenges».

[14] G. Delugré, «Evolution of 3GPP over-the-air security,» 10 May 2018. [En línea]. Available: https://gdelugre.github.io/2018/05/10/3gpp-ota-security-evolution/.

[15] ENISA, «Security in 5G Specifications,» 2021.

[16] ETSI, «Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; 5G; Numbering, addressing and identification».

[17] I. I. b. CableLabs, «A Comparative Introduction to 4G and 5G Authentication,» 2019. [En línea]. Available: https://www.cablelabs.com/insights/a-comparative-introduction-to-4g-and-5g-authentication.

[18] T. Y. A. D. MINISTERIO DE ENERGÍA, «Plan Nacional 5G 2018-2020,» 2018.

[19] Josh, «BANDANCHA.EU,» 7 enero 2021. [En línea]. Available: https://bandaancha.eu/articulos/frecuencias-5g-operadoras-espana-9815.

[20] «Crowd Supply,» [En línea]. Available: https://www.crowdsupply.com/lime-micro/limesdr.

[21] [En línea]. Available: http://shop.sysmocom.de/products/sysmoISIM-SJA2.

[22] H. Welte, *sysmoUSIM / sysmoISIM User Manual,* 2021.

[23] «Características técnicas Oppo Reno 4 Z 5G,» [En línea]. Available: https://www.smart-gsm.com/moviles/oppo-reno-4-z-5g.

[24] « OPENAIRINTERFACE / openair-epc-fed - GITHUB,» [En línea]. Available: https://github.com/OPENAIRINTERFACE/openair-epc-fed/blob/2021.w06/docs/DEPLOY_HOME.md.

[25] «Docker Hub,» [En línea]. Available: https://hub.docker.com/.

[26] «Open Air Interface (OAI) - oai-cn5g-fed,» [En línea]. Available: https://gitlab.eurecom.fr/oai/cn5g/oai-cn5g-fed/-/blob/master/docs/DEPLOY_HOME.md.

[27] P. Sahu, «5G Resource Center Blogs,» 15 May 2019. [En línea]. Available: http://5gblogs.com/concealing-of-supi-into-suci/.

[28] «Ettus Research,» [En línea]. Available: https://www.ettus.com/.

[29] 3GPP, «System Architecture for 5G System,» 2018.

[30] 3GPP, «Procedures for the 5G System - Technical Specification,» 2018.

[31] «Amarisoft,» [En línea]. Available: https://www.amarisoft.com/.

[32] «YateBTS,» [En línea]. Available: https://yatebts.com/.

[33] «Nuand,» [En línea]. Available: https://www.nuand.com/bladerf-1/.

9. Apéndices

9.1. Objetivos de Desarrollo Sostenible – ODS

Los Objetivos de Desarrollo Sostenible son un conjunto de propósitos establecidos mundialmente para «erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible».

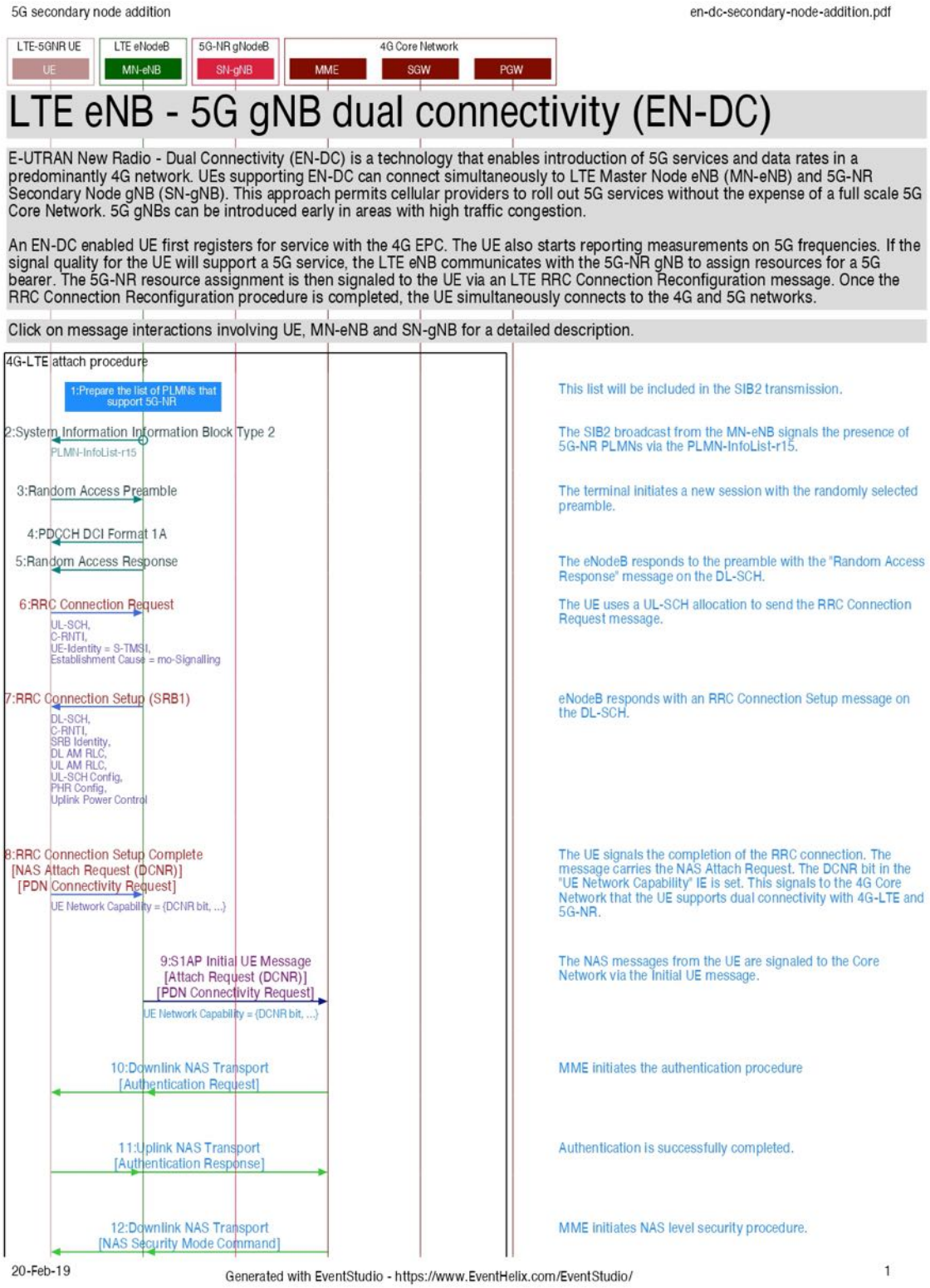
Este proyecto esta alineado con los siguientes objetivos:

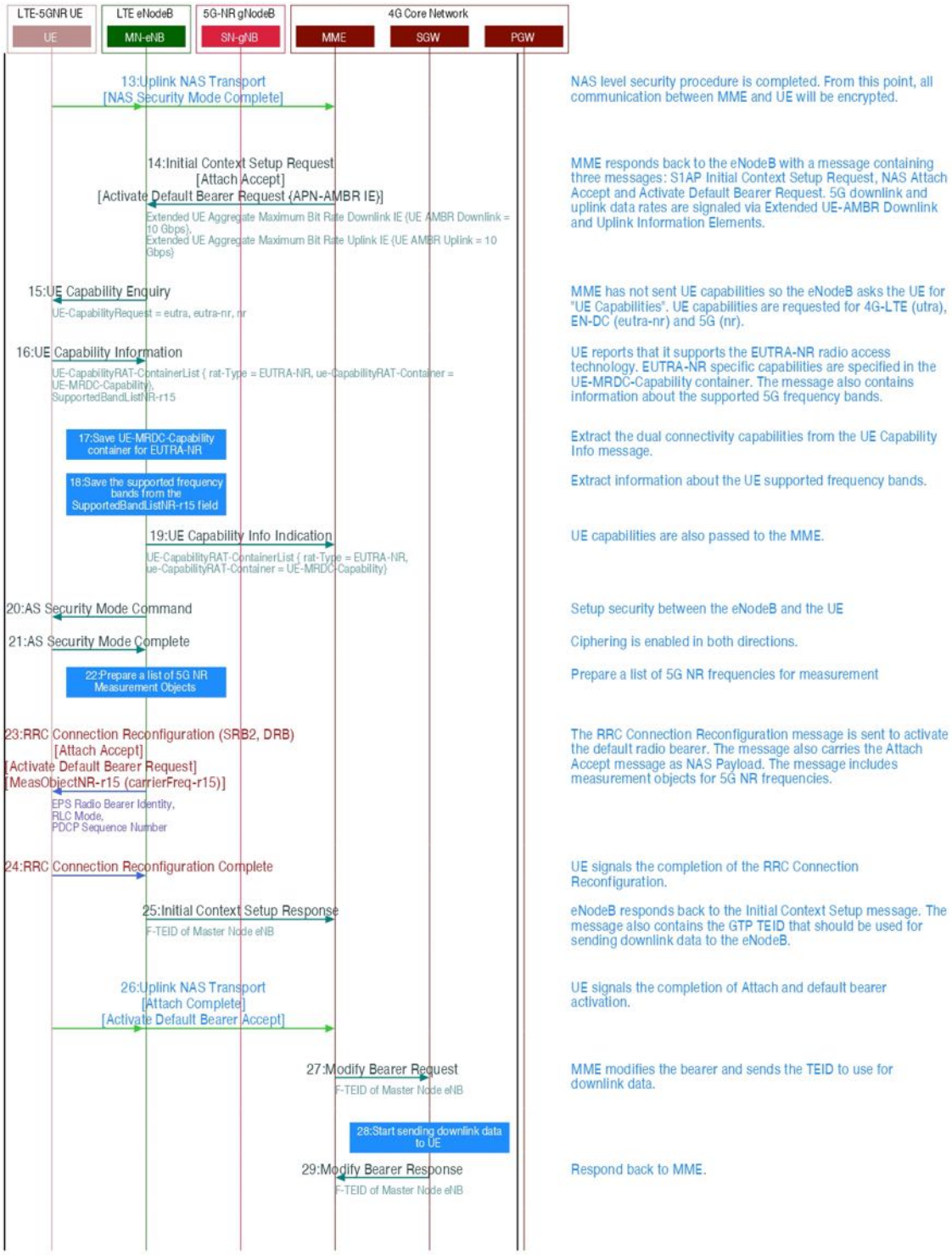
Educación de Calidad: Además de los objetivos concretos de este proyecto, también se va a fomentar el estudio de nuevas tecnologías como las redes móviles 5G. Asimismo, para el proyecto se va a utilizar software libre (open source) puesto que es una forma de fomentar el desarrollo académico, tanto individual como universitario. Uno de los claros ejemplos es el SDR (Software Defined Radio) que ha cambiado radicalmente el despliegue de redes móviles en laboratorios.

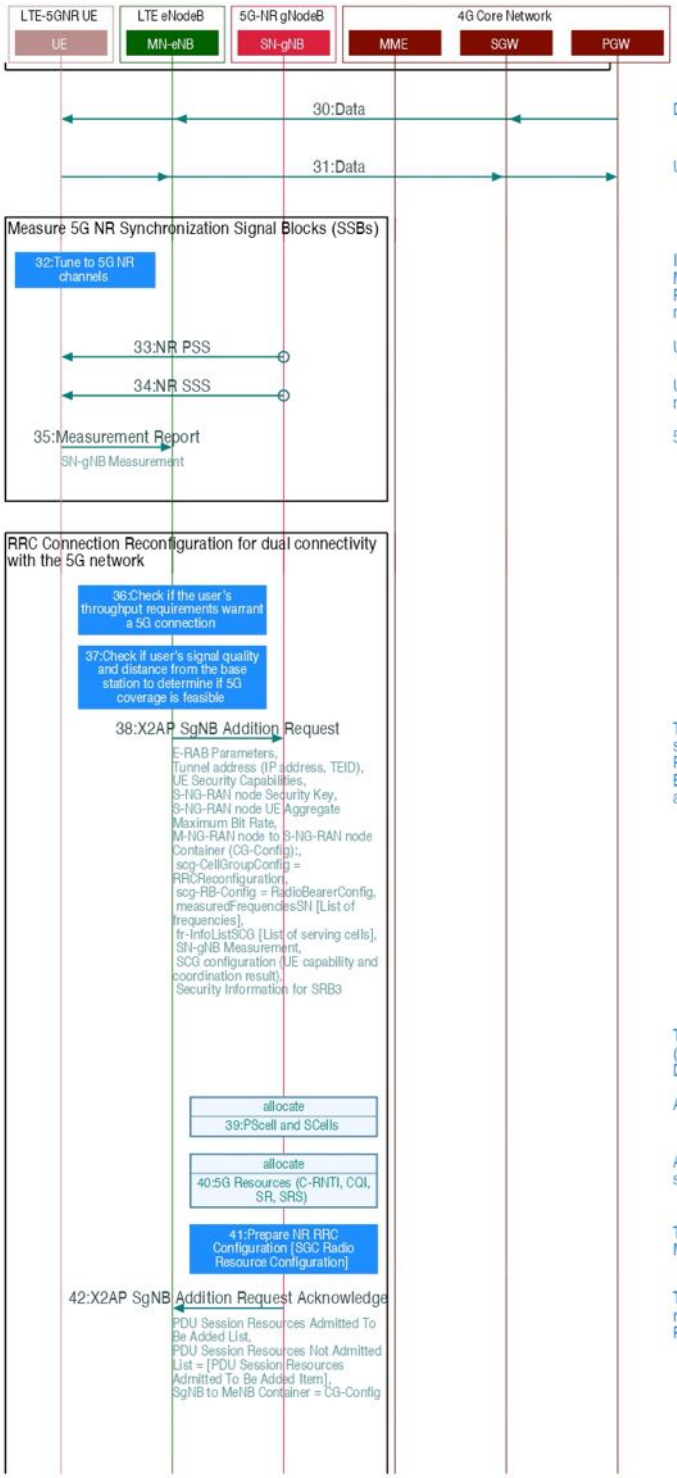
Industria, Innovación e Infraestructura + Ciudades y comunidades sostenibles: El despliegue de las redes 5G van a producir un cambio significativo en la forma de comunicarse no solo de los individuos, pero también de las máquinas. La velocidad proporcionada por esta tecnología permitirá el desarrollo de ciudades inteligentes y comunidades conectadas.

9.2. Procedimientos de Acople 5G

9.2.1. NSA




LTE-5GNR UE
UE
LTE eNodeB
MN-eNB
5G-NR gNodeB
SN-gNB
4G Core Network
MME
SGW
PGW
13:Uplink NAS Transport
[NAS Security Mode Complete]
NAS level security procedure is completed. From this point, all communication between MME and UE will be encrypted.
14:Initial Context Setup Request
[Attach Accept]
[Activate Default Bearer Request (APN-AMBR IE)]
Extended UE Aggregate Maximum Bit Rate Downlink IE (UE AMBR Downlink = 10 Gbps),
Extended UE Aggregate Maximum Bit Rate Uplink IE (UE AMBR Uplink = 10 Gbps)
MME responds back to the eNodeB with a message containing three messages: S1AP Initial Context Setup Request, NAS Attach Accept and Activate Default Bearer Request. 5G downlink and uplink data rates are signaled via Extended UE-AMBR Downlink and Uplink Information Elements.
15:UE Capability Enquiry
UE-CapabilityRequest = eutra, eutra-nr, nr
MME has not sent UE capabilities so the eNodeB asks the UE for "UE Capabilities". UE capabilities are requested for 4G-LTE (utra), EN-DC (eutra-nr) and 5G (nr).
16:UE Capability Information
UE-CapabilityRAT-ContainerList { rat-Type = EUTRA-NR, ue-CapabilityRAT-Container = UE-MRDC-Capability},
SupportedBandListNR-r15
UE reports that it supports the EUTRA-NR radio access technology. EUTRA-NR specific capabilities are specified in the UE-MRDC-Capability container. The message also contains information about the supported 5G frequency bands.
17:Save UE-MRDC-Capability container for EUTRA-NR
Extract the dual connectivity capabilities from the UE Capability Info message.
18:Save the supported frequency bands from the SupportedBandListNR-r15 field
Extract information about the UE supported frequency bands.
19:UE Capability Info Indication
UE-CapabilityRAT-ContainerList { rat-Type = EUTRA-NR, ue-CapabilityRAT-Container = UE-MRDC-Capability}
UE capabilities are also passed to the MME.
20:AS Security Mode Command
Setup security between the eNodeB and the UE
21:AS Security Mode Complete
Ciphering is enabled in both directions.
22:Prepare a list of 5G NR Measurement Objects
Prepare a list of 5G NR frequencies for measurement
23:RRC Connection Reconfiguration (SRB2, DRB)
[Attach Accept]
[Activate Default Bearer Request]
[MeasObjectNR-r15 (carrierFreq-r15)]
EPS Radio Bearer Identity,
RLC Mode,
PDCP Sequence Number
The RRC Connection Reconfiguration message is sent to activate the default radio bearer. The message also carries the Attach Accept message as NAS Payload. The message includes measurement objects for 5G NR frequencies.
24:RRC Connection Reconfiguration Complete
UE signals the completion of the RRC Connection Reconfiguration.
25:Initial Context Setup Response
F-TEID of Master Node eNB
eNodeB responds back to the Initial Context Setup message. The message also contains the GTP TEID that should be used for sending downlink data to the eNodeB.
26:Uplink NAS Transport
[Attach Complete]
[Activate Default Bearer Accept]
UE signals the completion of Attach and default bearer activation.
27:Modify Bearer Request
F-TEID of Master Node eNB
MME modifies the bearer and sends the TEID to use for downlink data.
28:Start sending downlink data to UE
29:Modify Bearer Response
F-TEID of Master Node eNB
Respond back to MME.



Downlink data is flowing on the default bearer.

Uplink data is flowing on the default bearer.

Initiate measurement of 5G NR channels specified in the Measurement Objects received in the RRC Connection Reconfiguration. These measurements are scheduled during measurement gaps.

UE acquires the 5G-NR Primary Synchronization Signal.

UE acquires the 5G-NR Secondary Synchronization Signal and measures the signal quality.

5G signal quality is reported back to 4G eNB.

The 4G LTE eNodeB decides to add the 5G-NR base station as a secondary node. The eNodeB sends a Secondary Node Addition Request to the gNodeB. The message carries the RRC and Radio Bearer configuration. UE capabilities and security information are also included in the message.

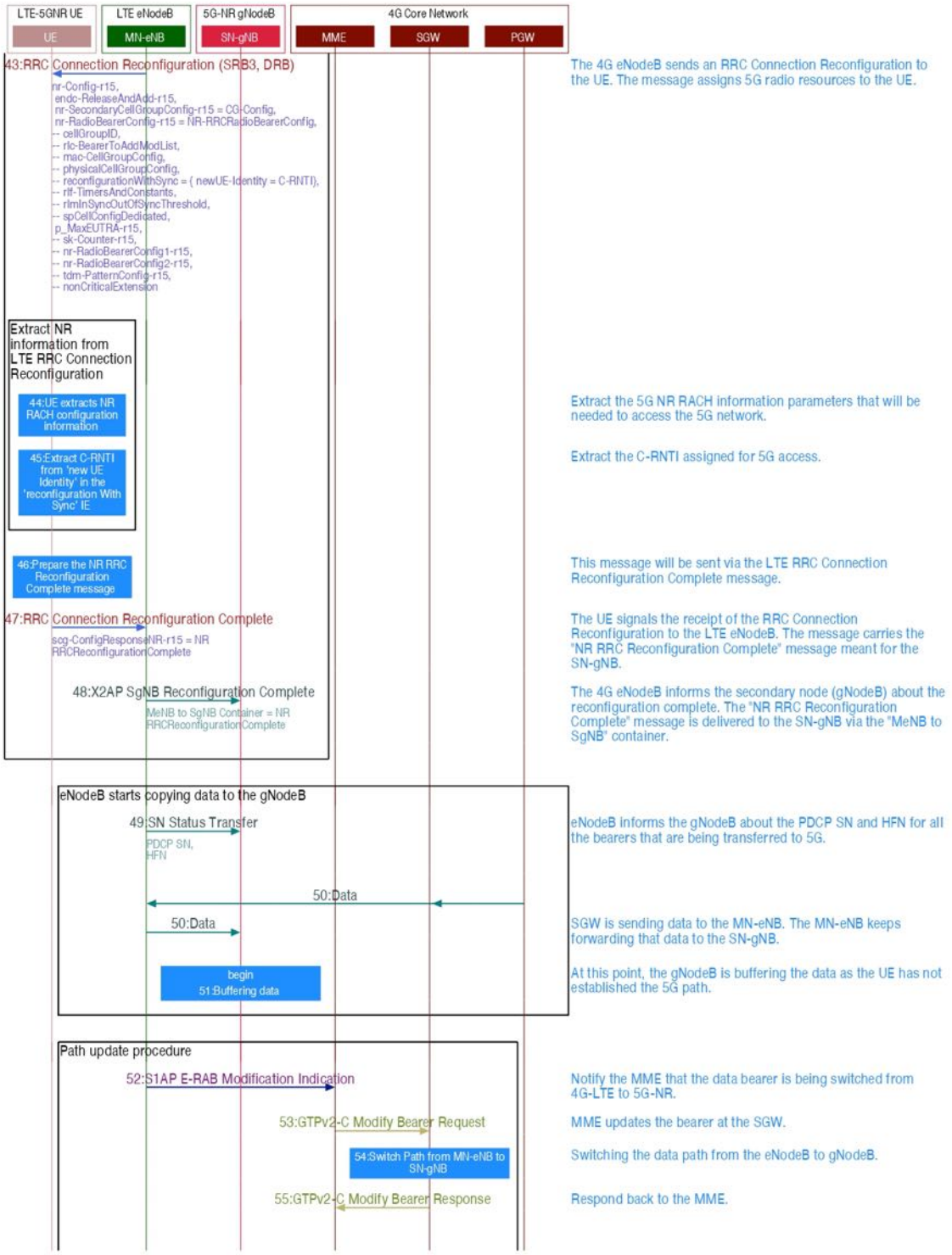
The network indicates whether the UE shall use either KeNB (master node key) or S-KgNB (secondary node key) for the 5G DRB.

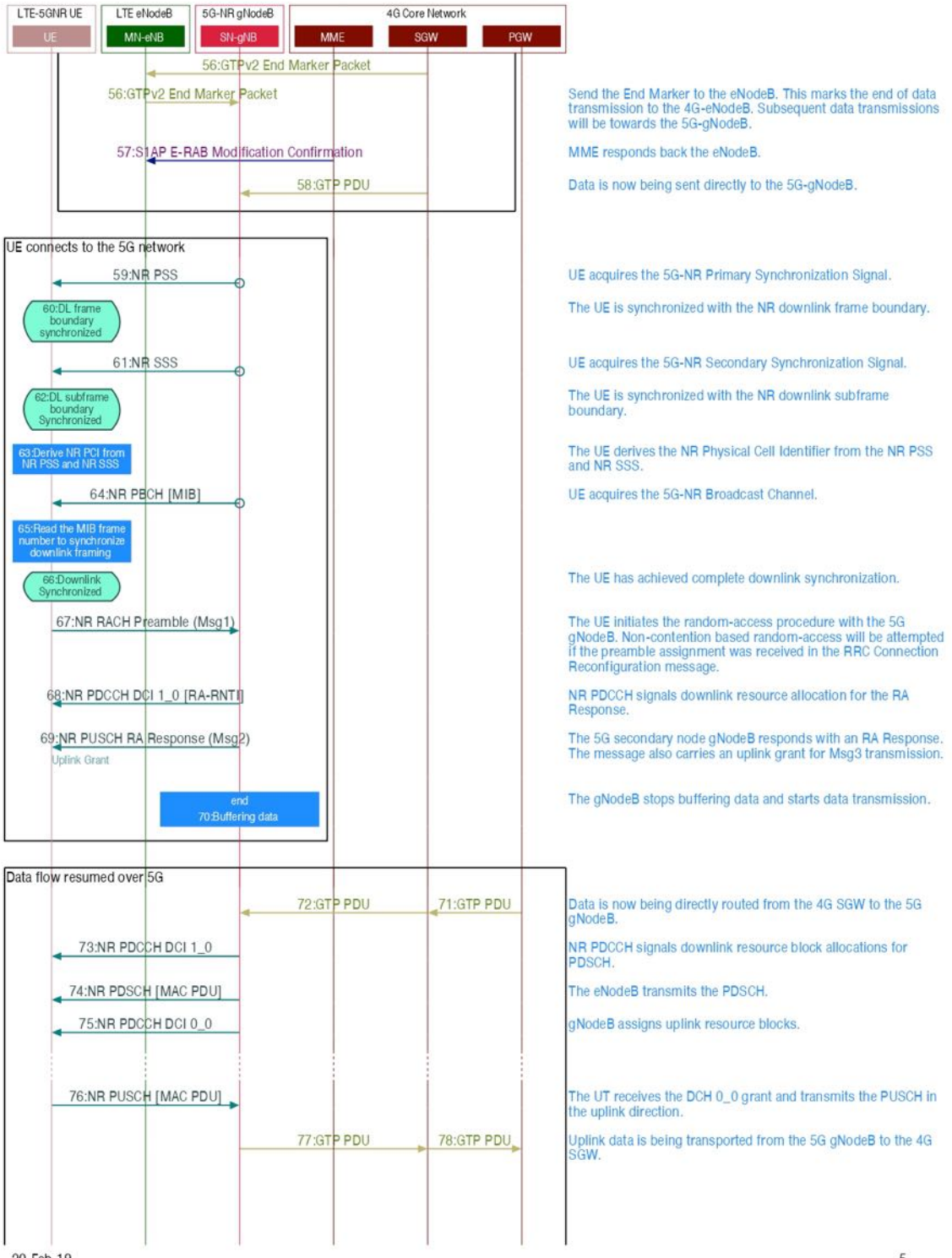
Allocate 5G cells

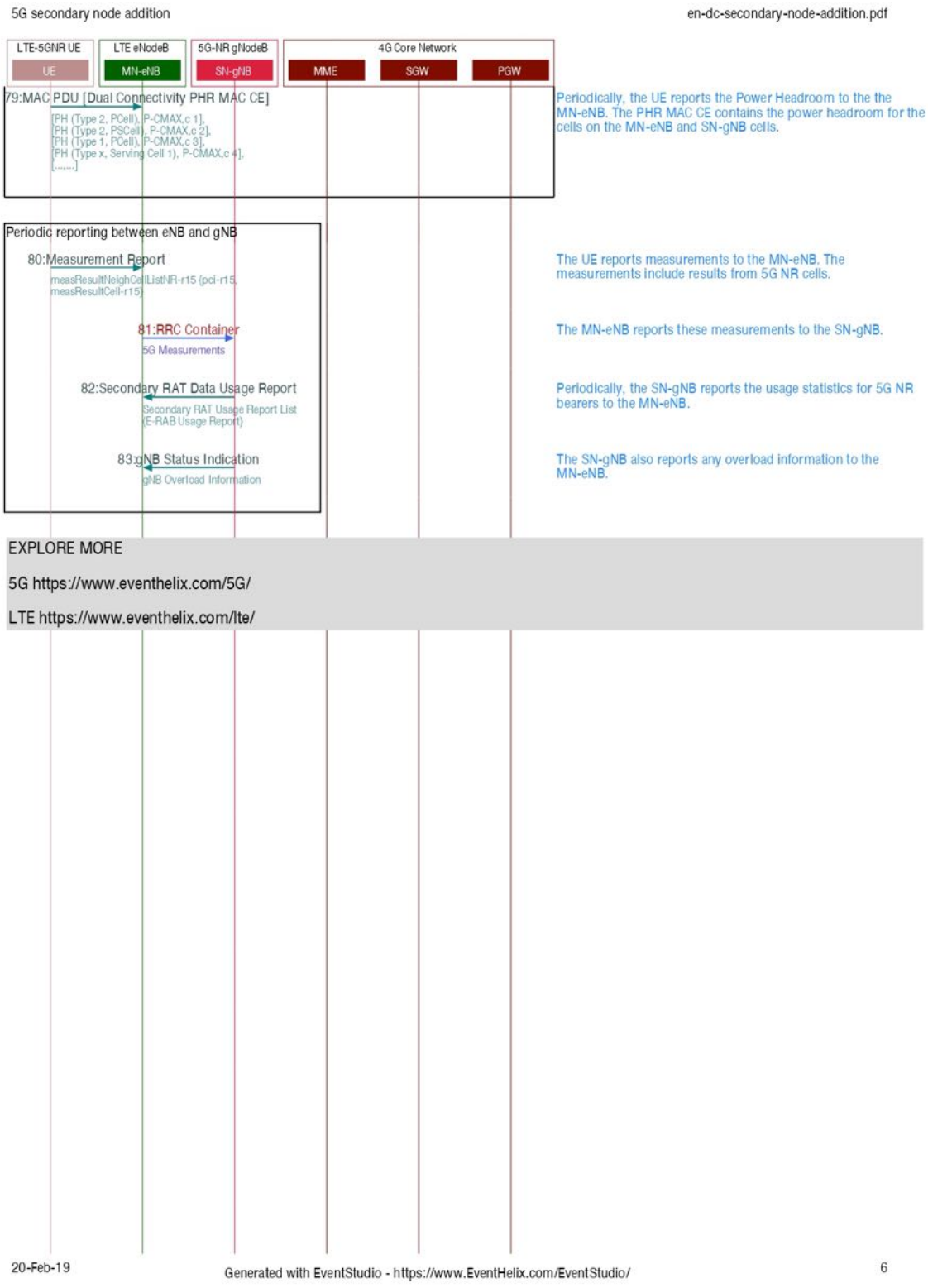
Allocate the 5G radio resources needed for the secondary session.

The NR RRC Configuration will be transmitted to the UE via the MN-eNB.

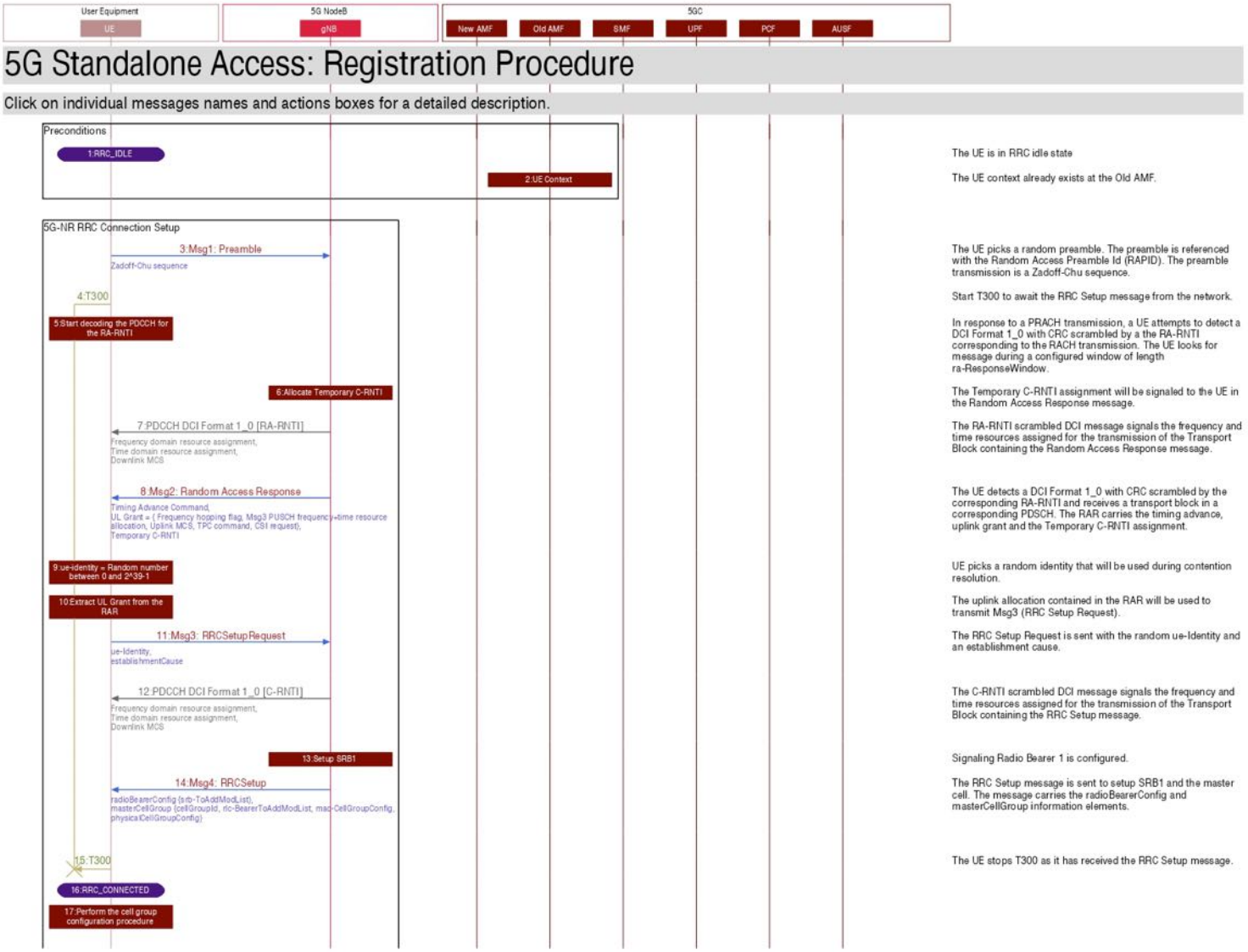
The gNodeB responds with information about the radio resources and bearers admitted with the 5G network. The NR RRC configuration message is included in the message.


LTE-5GNR UE
UE
LTE eNodeB
MN-eNB
5G-NR gNodeB
SN-gNB
4G Core Network
MME
SGW
PGW
43:RRC Connection Reconfiguration (SRB3, DRB)
nr-Config-r15,
endc-ReleaseAndAdd-r15,
nr-SecondaryCellGroupConfig-r15 = CG-Config,
nr-RadioBearerConfig-r15 = NR-RRCRadioBearerConfig,
-- cellGroupID,
-- rlc-BearerToAddModList,
-- mac-CellGroupConfig,
-- physicalCellGroupConfig,
-- reconfigurationWithSync = { newUE-Identity = C-RNTI},
-- rlf-TimersAndConstants,
-- rlmInSyncOutOfSyncThreshold,
-- spCellConfigDedicated,
p_MaxEUTRA-r15,
-- sk-Counter-r15,
-- nr-RadioBearerConfig1-r15,
-- nr-RadioBearerConfig2-r15,
-- tdm-PatternConfig-r15,
-- nonCriticalExtension
The 4G eNodeB sends an RRC Connection Reconfiguration to the UE. The message assigns 5G radio resources to the UE.
Extract NR information from LTE RRC Connection Reconfiguration
44:UE extracts NR RACH configuration information
Extract the 5G NR RACH information parameters that will be needed to access the 5G network.
45:Extract C-RNTI from "new UE Identity" in the "reconfiguration With Sync" IE
Extract the C-RNTI assigned for 5G access.
46:Prepare the NR RRC Reconfiguration Complete message
This message will be sent via the LTE RRC Connection Reconfiguration Complete message.
47:RRC Connection Reconfiguration Complete
scg-ConfigResponseNR-r15 = NR RRCReconfigurationComplete
The UE signals the receipt of the RRC Connection Reconfiguration to the LTE eNodeB. The message carries the "NR RRC Reconfiguration Complete" message meant for the SN-gNB.
48:X2AP SgNB Reconfiguration Complete
MeNB to SgNB Container = NR RRCReconfigurationComplete
The 4G eNodeB informs the secondary node (gNodeB) about the reconfiguration complete. The "NR RRC Reconfiguration Complete" message is delivered to the SN-gNB via the "MeNB to SgNB" container.
eNodeB starts copying data to the gNodeB
49:SN Status Transfer
PDCP SN, HFN
eNodeB informs the gNodeB about the PDCP SN and HFN for all the bearers that are being transferred to 5G.
50:Data
50:Data
SGW is sending data to the MN-eNB. The MN-eNB keeps forwarding that data to the SN-gNB.
begin 51:Buffering data
At this point, the gNodeB is buffering the data as the UE has not established the 5G path.
Path update procedure
52:S1AP E-RAB Modification Indication
Notify the MME that the data bearer is being switched from 4G-LTE to 5G-NR.
53:GTPv2-C Modify Bearer Request
MME updates the bearer at the SGW.
54:Switch Path from MN-eNB to SN-gNB
Switching the data path from the eNodeB to gNodeB.
55:GTPv2-C Modify Bearer Response
Respond back to the MME.


LTE-5GNR UE
UE
LTE eNodeB
MN-eNB
5G-NR gNodeB
SN-gNB
4G Core Network
MME
SGW
PGW
56:GTPv2 End Marker Packet
56:GTPv2 End Marker Packet
57:S1AP E-RAB Modification Confirmation
58:GTP PDU
UE connects to the 5G network
59:NR PSS
60:DL frame boundary synchronized
61:NR SSS
62:DL subframe boundary Synchronized
63:Derive NR PCI from NR PSS and NR SSS
64:NR PBCH [MIB]
65:Read the MIB frame number to synchronize downlink framing
66:Downlink Synchronized
67:NR RACH Preamble (Msg1)
68:NR PDCCH DCI 1_0 [RA-RNTI]
69:NR PUSCH RA Response (Msg2)
Uplink Grant
end
70:Buffering data
Data flow resumed over 5G
72:GTP PDU
71:GTP PDU
73:NR PDCCH DCI 1_0
74:NR PDSCH [MAC PDU]
75:NR PDCCH DCI 0_0
76:NR PUSCH [MAC PDU]
77:GTP PDU
78:GTP PDU
Send the End Marker to the eNodeB. This marks the end of data transmission to the 4G-eNodeB. Subsequent data transmissions will be towards the 5G-gNodeB.
MME responds back the eNodeB.
Data is now being sent directly to the 5G-gNodeB.
UE acquires the 5G-NR Primary Synchronization Signal.
The UE is synchronized with the NR downlink frame boundary.
UE acquires the 5G-NR Secondary Synchronization Signal.
The UE is synchronized with the NR downlink subframe boundary.
The UE derives the NR Physical Cell Identifier from the NR PSS and NR SSS.
UE acquires the 5G-NR Broadcast Channel.
The UE has achieved complete downlink synchronization.
The UE initiates the random-access procedure with the 5G gNodeB. Non-contention based random-access will be attempted if the preamble assignment was received in the RRC Connection Reconfiguration message.
NR PDCCH signals downlink resource allocation for the RA Response.
The 5G secondary node gNodeB responds with an RA Response. The message also carries an uplink grant for Msg3 transmission.
The gNodeB stops buffering data and starts data transmission.
Data is now being directly routed from the 4G SGW to the 5G gNodeB.
NR PDCCH signals downlink resource block allocations for PDSCH.
The eNodeB transmits the PDSCH.
gNodeB assigns uplink resource blocks.
The UT receives the DCH 0_0 grant and transmits the PUSCH in the uplink direction.
Uplink data is being transported from the 5G gNodeB to the 4G SGW.


5G secondary node addition
en-dc-secondary-node-addition.pdf
LTE-5GNR UE
UE
LTE eNodeB
MN-eNB
5G-NR gNodeB
SN-gNB
4G Core Network
MME
SGW
PGW
79:MAC PDU [Dual Connectivity PHR MAC CE]
[PH (Type 2, PCell), P-CMAX,c 1],
[PH (Type 2, PSCell), P-CMAX,c 2],
[PH (Type 1, PCell), P-CMAX,c 3],
[PH (Type x, Serving Cell 1), P-CMAX,c 4],
[.....]
Periodically, the UE reports the Power Headroom to the the MN-eNB. The PHR MAC CE contains the power headroom for the cells on the MN-eNB and SN-gNB cells.
Periodic reporting between eNB and gNB
80:Measurement Report
measResultNeighCellListNR-r15 (pci-r15, measResultCell-r15)
The UE reports measurements to the MN-eNB. The measurements include results from 5G NR cells.
81:RRC Container
5G Measurements
The MN-eNB reports these measurements to the SN-gNB.
82:Secondary RAT Data Usage Report
Secondary RAT Usage Report List (E-RAB Usage Report)
Periodically, the SN-gNB reports the usage statistics for 5G NR bearers to the MN-eNB.
83:gNB Status Indication
gNB Overload Information
The SN-gNB also reports any overload information to the MN-eNB.
EXPLORE MORE
5G https://www.eventhelix.com/5G/
LTE https://www.eventhelix.com/lte/
20-Feb-19
Generated with EventStudio - https://www.EventHelix.com/EventStudio/
6

9.2.2. SA



5G Standalone Access: Registration Procedure

Click on individual messages names and actions boxes for a detailed description.

Preconditions

1:RRC_IDLE

2:UE Context

The UE is in RRC idle state

The UE context already exists at the Old AMF.

5G-NR RRC Connection Setup

3:Msg1: Preamble

Zadoff-Chu sequence

The UE picks a random preamble. The preamble is referenced with the Random Access Preamble Id (RAPID). The preamble transmission is a Zadoff-Chu sequence.

4:T300

Start T300 to await the RRC Setup message from the network.

5:Start decoding the PDCCH for the RA-RNTI

In response to a PRACH transmission, a UE attempts to detect a DCI Format 1_0 with CRC scrambled by a the RA-RNTI corresponding to the RACH transmission. The UE looks for message during a configured window of length ra-ResponseWindow.

6:Allocate Temporary C-RNTI

The Temporary C-RNTI assignment will be signaled to the UE in the Random Access Response message.

7:PDCCH DCI Format 1_0 [RA-RNTI]

Frequency domain resource assignment,
Time domain resource assignment,
Downlink MCS

The RA-RNTI scrambled DCI message signals the frequency and time resources assigned for the transmission of the Transport Block containing the Random Access Response message.

8:Msg2: Random Access Response

Timing Advance Command,
UL Grant = { Frequency hopping flag, Msg3 PUSCH frequency-time resource allocation, Uplink MCS, TPC command, CSI request};
Temporary C-RNTI

The UE detects a DCI Format 1_0 with CRC scrambled by the corresponding RA-RNTI and receives a transport block in a corresponding PDSCH. The RAR carries the timing advance, uplink grant and the Temporary C-RNTI assignment.

9:ue-identity = Random number between 0 and 2^39-1

UE picks a random identity that will be used during contention resolution.

10:Extract UL Grant from the RAR

The uplink allocation contained in the RAR will be used to transmit Msg3 (RRC Setup Request).

11:Msg3: RRCSetupRequest

ue-Identity,
establishmentCause

The RRC Setup Request is sent with the random ue-Identity and an establishment cause.

12:PDCCH DCI Format 1_0 [C-RNTI]

Frequency domain resource assignment,
Time domain resource assignment,
Downlink MCS

The C-RNTI scrambled DCI message signals the frequency and time resources assigned for the transmission of the Transport Block containing the RRC Setup message.

13:Setup SRB1

Signaling Radio Bearer 1 is configured.

14:Msg4: RRCSetup

radioBearerConfig {srb-ToAddModList},
masterCellGroup {cellGroupId, rlc-BearerToAddModList, mac-CellGroupConfig, physicalCellGroupConfig}

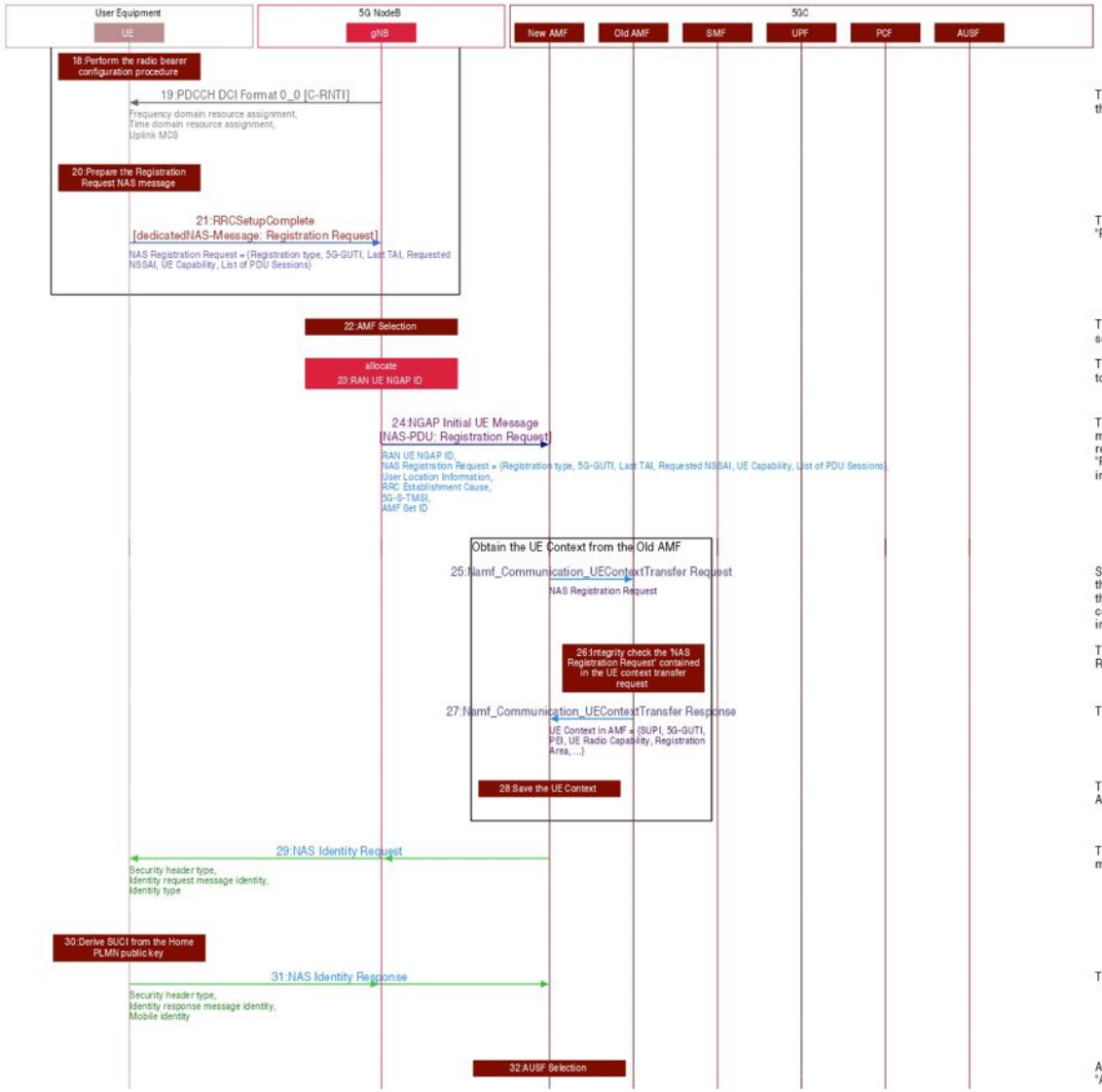
The RRC Setup message is sent to setup SRB1 and the master cell. The message carries the radioBearerConfig and masterCellGroup information elements.

15:T300

The UE stops T300 as it has received the RRC Setup message.

16:RRC_CONNECTED

17:Perform the cell group configuration procedure



The gNB assigns uplink resource to the UE so that it can send the RRC Setup Complete message.

The UE sends the RRC Setup Complete message with a "Registration Request" in the dedicatedNAS-Message field.

The gNB selects the Access and Mobility Function (AMF) for this session.

The gNB allocates a "RAN UE NGAP ID". The AMF will use this id to address the UE context on the gNB.

The gNB sends the Initial UE Message to the selected AMF. The message carries the "Registration Request" message that was received from the UE in the RRC Setup Complete message. The "RAN UE NGAP ID" and the "RRC Establishment Cause" are also included in the message.

Since the 5G-GUTI was included in the Registration Request and the serving AMF has changed since last Registration procedure, the new AMF requests context transfer from the old AMF. The complete NAS registration message received from the UE is included in the context request.

The AMF performs an integrity check on the Registration Request to guard against malicious attacks.

The Old AMF passes the AMF UE Context to the new AMF.

The AMF saves the UE context that was obtained from the Old AMF.

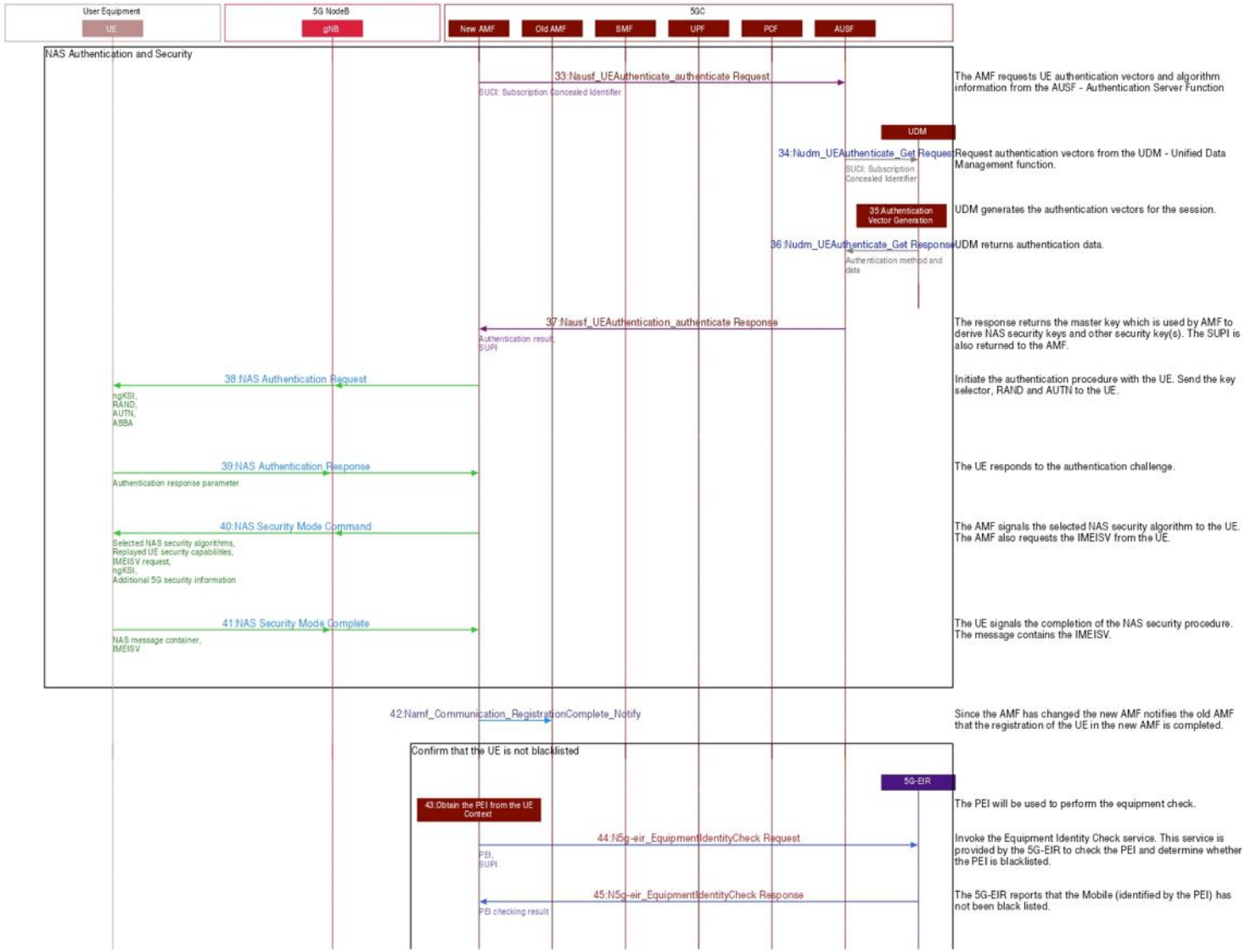
The New AMF requests UE Identity (SUCI) from the UE via a NAS message.

The UE responds to the Identity Request.

Authentication is needed for the UE. The AMF selects "Authentication Server Function" based on the SUCI.

5G Standalone Access Registration

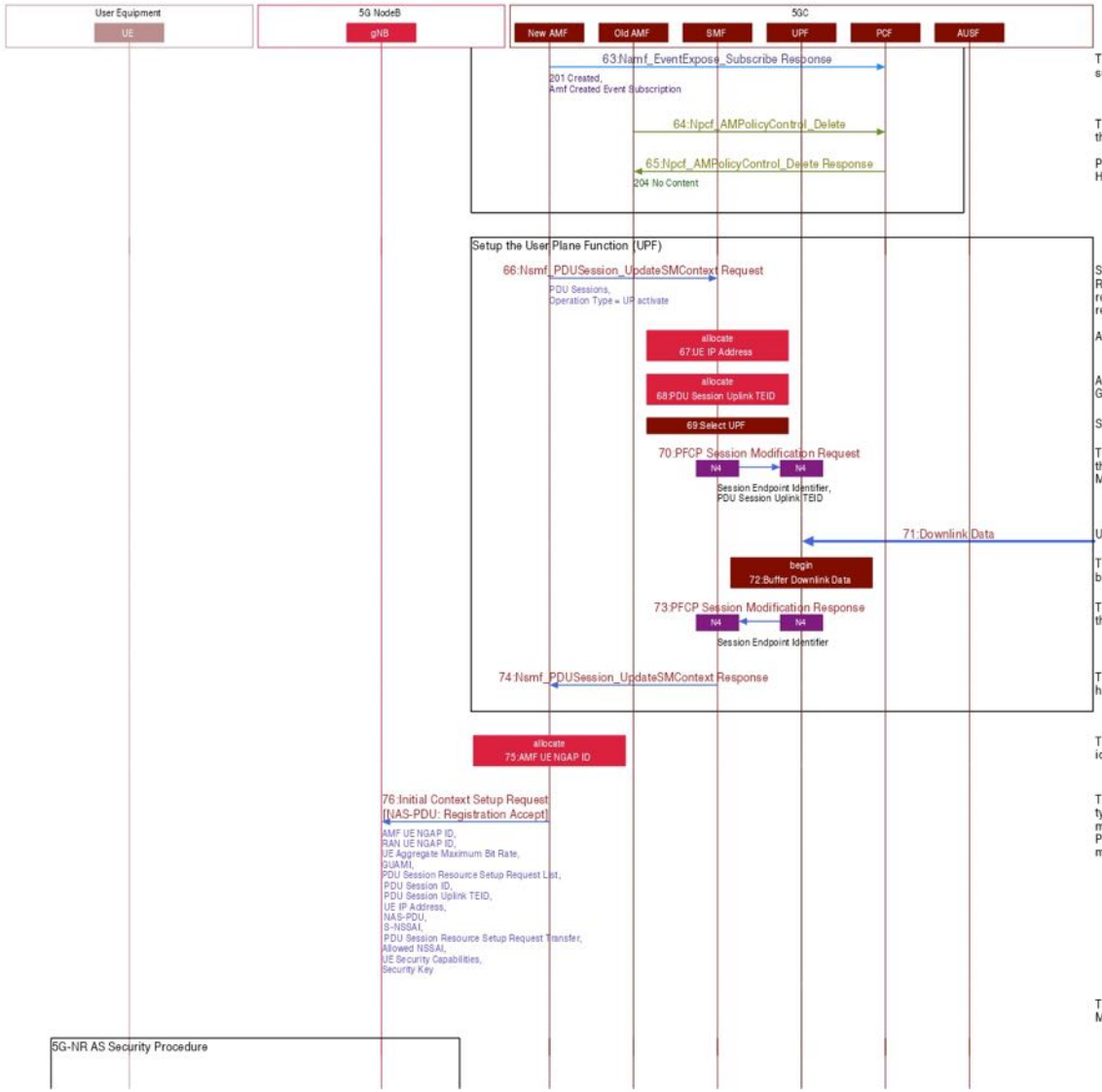
5g-standalone-access-registration.pdf




User Equipment
UE
5G NodeB
gNB
5GC
New AMF
Old AMF
SMF
UPF
PCF
AUSF
5G-EIR
Register with the UDM and obtain the subscription data
46:UDM Selection
Select the Unified Data Management service entity.
UDM
47:Nudm_UEContextManagement_Registration Request
PUT,
Amf 3Gpp Access Registration = { AMF Instance Id, Supported Features, PEI, dereg Callback Uri, ...}
Since the AMF has changed, the New AMF registers with the UDM.
48:Nudm_UEContextManagement_Registration Response
204 No Content
A response code of "204 No Content" signals registration success.
49:Nudm_SubscriberDataManagement_Get Request
GET,
Requested data = Access and Mobility Subscription data
The AMF retrieves the Access and Mobility Subscription data.
50:Nudm_SubscriberDataManagement_Get Response
Access and Mobility Subscription data = {Supported Features, GPSI array, Network Slice Selection Info, ...}
UDM responds with the requested data.
51:Nudm_SubscriberDataManagement_Get Request
GET,
Requested data = SMF Selection Subscription data
The AMF retrieves the SMF Selection Subscription data.
52:Nudm_SubscriberDataManagement_Get Response
SMF Selection Subscription data = {Supported Features, List of S-NSSAIs and associated information}
UDM responds with the requested data.
53:Nudm_SubscriberDataManagement_Get Request
GET,
Requested data = UE context in SMF data
The AMF retrieves the UE context in SMF data.
54:Nudm_SubscriberDataManagement_Get Response
UE context in SMF data = {PDU Session Information, FQDNs for EPC Interworking}
UDM responds with the requested data.
55:Create UE Context
Create the AMF UE context for the user.
56:Nudm_UEContextManagement_Deregistration_Notify
The old AMF is notified that it is no longer serving the user. The UDM sends a POST request to the callbackReference (deregCallbackUri field in Amf3GppAccessRegistration) as provided by the Old AMF during the registration.
57:Nsmf_PDUSession_ReleaseSMContext
SUPI,
PDU Session ID
The Old AMF the SMF that it is no longer associated with the specified PDU Session.
58:Delete UE context
Remove the AMF context.
Update policy association with the PCF. The PCF registers for AMF events.
59:PCF Selection
Select the Policy and Charging Function service entity.
60:Npcf_AMPolicyControl_Create Request
The AMF contacts the PCF to create a policy association and retrieve the UE policy and/or Access and Mobility control policy.
61:Npcf_AMPolicyControl_Create Response
Policy Association Request = { SUPI, GPSI, PEI, User Location, ...}
The PCF responds with the policy association information.
62:Namf_EventExpose_Subscribe Request
The PCF registers for events like "Location Report", "Registration State Report" and "Communication Failure Report".

5G Standalone Access Registration

5g-standalone-access-registration.pdf



The AMF responds with "201 Created" to signal successful subscription.

The Old AMF requests that the policy association is deleted as the corresponding UE context is terminated.

PCF signals the successful delete with the "204 No Content" HTTP response code.

Since "List Of PDU Sessions To Be Activated" was included in the Registration Request, the New AMF initiates PDU Session reactivation. The Session Management Function (SMF) is requested to setup a new session.

Assign a UE address for the PDU session

Allocate a TEID that the gNB should use when sending uplink GTP PDUs to the UPF.

Select a user data plane for the user

The Packet Forwarding Control Protocol (PFCP) is used between the SMF control plane and the UPF data plane. The Session Modification is signaled to the data plane.

UPF has started receiving data destined to the UE.

The UPF needs to buffer the data as the PDU session has not been established at the gNB and the UE.

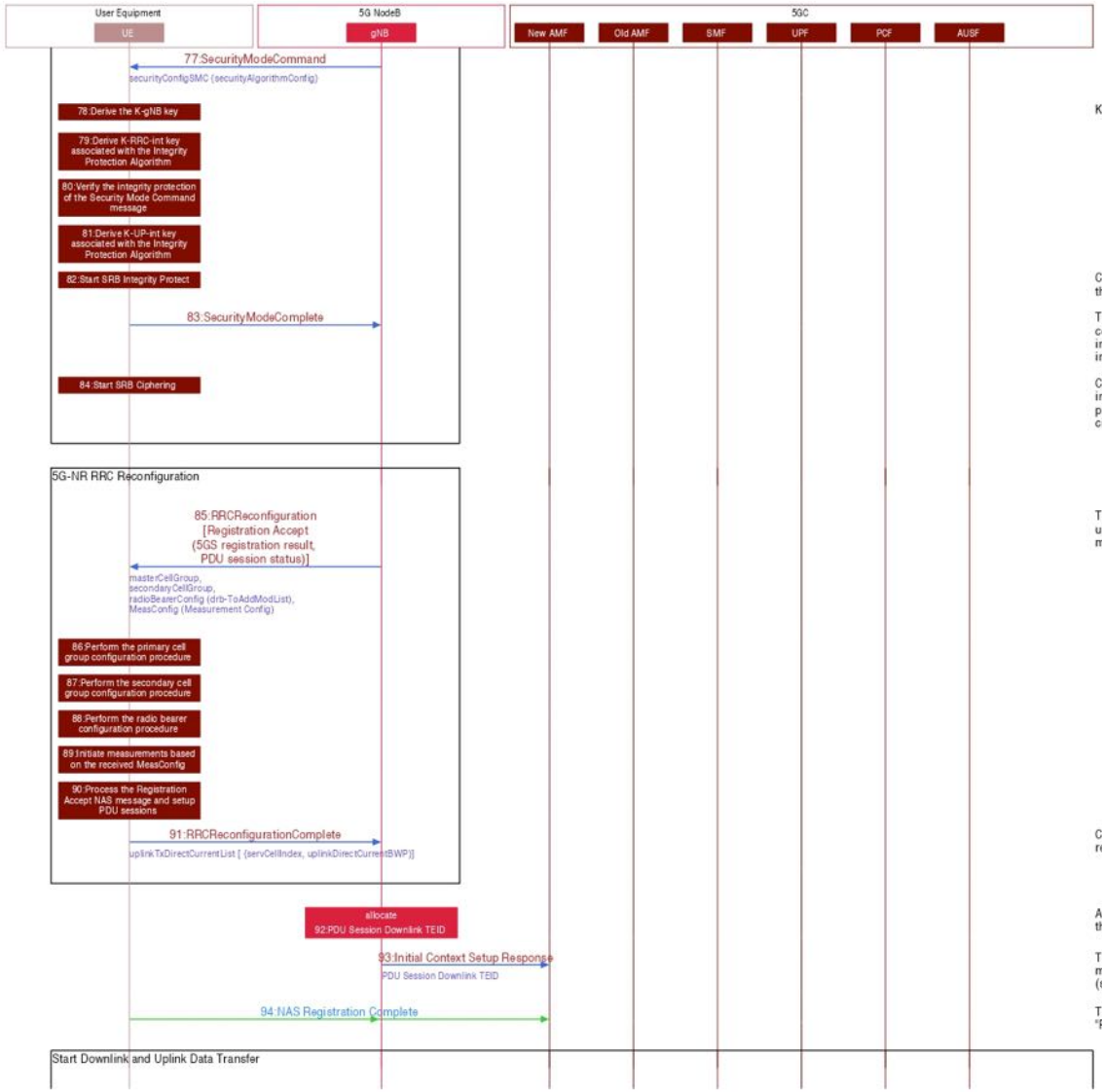
The UPF data plane responds back to the SMF control plane after the session modification has been completed.

The SMF informs the AMF that the Session Management context has been updated.

The AMF allocates an "AMF UE NGAP ID". The gNB will use this id to address the UE context on the AMF.

The AMF initiates a session setup with the gNB. The message typically contains the Registration Accept NAS message. The message carries one or more PDU Session setup requests. Each PDU session is addressed with the "PDU Session ID". The message also carries the uplink TEID for every PDU session.

The message also carries the "AMF UE NGAP ID", "UE Aggregate Maximum Bit Rate", UE security capabilities and security key.



K-gNB is a key derived by UE and AMF from K-AMF.

Configure lower layers to apply SRB integrity protection using the indicated algorithm and the K-RRC-int key immediately.

The security mode complete message confirms the successful completion of the security mode command. This message is integrity protected but not ciphered. Ciphering will start immediately after sending this message.

Configure lower layers to apply SRB ciphering using the indicated algorithm, the K-RRC-enc key after completing the procedure. The Security Mode Complete message is not ciphered.

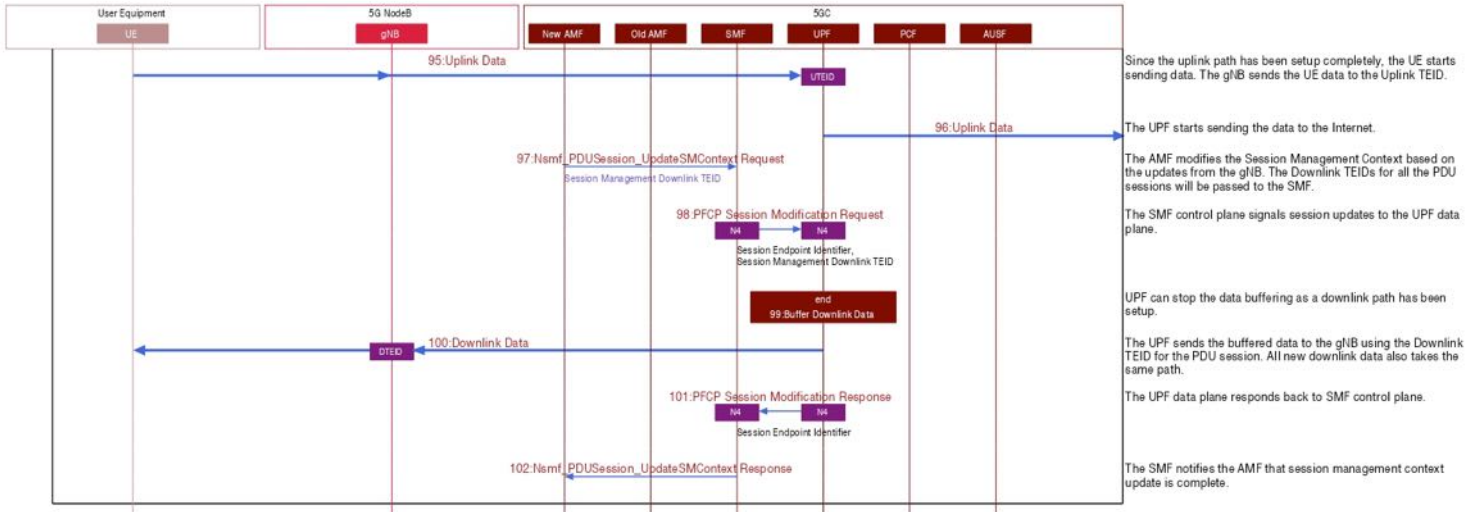
The RRC Reconfiguration message is sent to the UE for setting up radio bearers, setup a secondary cell and initiate UE measurements.

Confirm the successful completion of an RRC connection reconfiguration.

Allocate the TEID that the UPF will use to send downlink data to the gNB.

The gNB signals the successful setup of PDU sessions. The message also carries the Downlink TEID that should be used (specified per PDU session).

The UE signals the completion of the registration via the "Registration Complete" message to the AMF.



9.3. Instalación del software OAI

9.3.1. UHD

```
$ sudo apt-get install libuhd-dev libuhd003 uhd-host
```

Con esto podremos utilizar dos comandos relevantes: uhd_find_devices y uhd_usrp_probe

```
miguel@miguel-NUC10i7FNH:~/Documents/OAI/progrom_sim_cards/sysmo-usim-tool$
sudo uhd_find_devices

[INFO] [UHD] linux; GNU C++ version 7.5.0; Boost_106501; UHD_3.15.0.0-
release
[INFO] [B200] Loading firmware image:
/usr/share/uhd/images/usrp_b200_fw.hex...
-------------------------------------------------
-- UHD Device 0
-------------------------------------------------
Device Address:
    serial: F5EA8B
    name:
    product: B200
    type: b200
```

```
miguel@miguel-NUC10i7FNH:~/Documents/OAI/progrom_sim_cards/sysmo-usim-tool$
sudo uhd_usrp_probe

[INFO] [UHD] linux; GNU C++ version 7.5.0; Boost_106501; UHD_3.15.0.0-
release
[INFO] [B200] Detected Device: B200
[INFO] [B200] Loading FPGA image:
/usr/share/uhd/images/usrp_b200_fpga.bin...
[INFO] [B200] Operating over USB 3.
[INFO] [B200] Detecting internal GPSDO....
[INFO] [GPS] Found an internal GPSDO: GPSTCXO , Firmware Rev 0.929a
[INFO] [B200] Initialize CODEC control...
[INFO] [B200] Initialize Radio control...
[INFO] [B200] Performing register loopback test...
[INFO] [B200] Register loopback test passed
[INFO] [B200] Setting master clock rate selection to 'automatic'.
[INFO] [B200] Asking for clock rate 16.000000 MHz...
[INFO] [B200] Actually got clock rate 16.000000 MHz.
  _____________________________________________________
 /
|       Device: B-Series Device
|     _____________________________________________________
|    /
|   |       Mboard: B200
|   |   serial: F5EA8B
|   |   product: 1
|   |   revision: 4
|   |   FW Version: 8.0
|   |   FPGA Version: 16.0
|   |
|   |   Time sources:  none, internal, external, gpsdo
```

```
|    |   Clock sources: internal, external, gpsdo
|    |   Sensors: gps_gpgga, gps_gprmc, gps_time, gps_locked, gps_servo,
ref_locked
|    |     ____________________________________________________
|    |    /
|    |   |       RX DSP: 0
|    |   |
|    |   |   Freq range: -8.000 to 8.000 MHz
|    |     ____________________________________________________
|    |    /
|    |   |       RX Dboard: A
|    |   |     ______________________________________________________
|    |   |    /
|    |   |   |       RX Frontend: A
|    |   |   |   Name: FE-RX2
|    |   |   |   Antennas: TX/RX, RX2
|    |   |   |   Sensors: temp, rssi, lo_locked
|    |   |   |   Freq range: 50.000 to 6000.000 MHz
|    |   |   |   Gain range PGA: 0.0 to 76.0 step 1.0 dB
|    |   |   |   Bandwidth range: 200000.0 to 56000000.0 step 0.0 Hz
|    |   |   |   Connection Type: IQ
|    |   |   |   Uses LO offset: No
|    |   |     ______________________________________________________
|    |   |    /
|    |   |   |       RX Codec: A
|    |   |   |   Name: B200 RX dual ADC
|    |   |   |   Gain Elements: None
|    |     ____________________________________________________
|    |    /
|    |   |       TX DSP: 0
|    |   |
|    |   |   Freq range: -8.000 to 8.000 MHz
|    |     ____________________________________________________
|    |    /
|    |   |       TX Dboard: A
|    |   |     ______________________________________________________
|    |   |    /
|    |   |   |       TX Frontend: A
|    |   |   |   Name: FE-TX2
|    |   |   |   Antennas: TX/RX
|    |   |   |   Sensors: temp, lo_locked
|    |   |   |   Freq range: 50.000 to 6000.000 MHz
|    |   |   |   Gain range PGA: 0.0 to 89.8 step 0.2 dB
|    |   |   |   Bandwidth range: 200000.0 to 56000000.0 step 0.0 Hz
|    |   |   |   Connection Type: IQ
|    |   |   |   Uses LO offset: No
|    |   |     ______________________________________________________
|    |   |    /
|    |   |   |       TX Codec: A
|    |   |   |   Name: B200 TX dual DAC
|    |   |   |   Gain Elements: None
```

9.3.2. EPC

Para la instalación del EPC se siguió la siguiente referencia: [24]

Es importante recalcar que el repositorio utilizado para el proyecto esta en el tag: ***2021.w06*** puesto que existen nuevas actualizaciones donde se configuran los contenedores de una forma alternativa.

En primer lugar, instalamos docker en nuestra máquina mediante los siguientes pasos:

1. Actualizar el repositorio apt-get e instalar los paquetes necesarios para poder utilizar un repositorio sobre HTTPS

```
sudo apt-get update

sudo apt-get install \
  apt-transport-https \
  ca-certificates \
  curl \
  gnupg \
  lsb-release
```

2. Añadir la clave GPG oficial de Docker

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

3. Actualizar los paquetes e instalar docker:

```
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

4. Comprobar que docker se ha instalado correctamente ejecutando su imagen hello-world

```
sudo docker run hello-world
```

5. Una vez instalado docker, nos creamos una cuenta en Docker Hub [25] y nos descargamos las siguientes imágenes:

```
$ docker login
$ docker pull ubuntu:bionic
$ docker pull cassandra:2.1
$ docker logout
```

En segundo lugar, descargarse open-air-epc:

1. Descargar la versión correspondiente a este proyecto *(existen versiones más recientes)*

Las versiones de los distintos nodos son las siguientes:

Nombre del contenedor	Rama	Versión
HSS	master	v1.1.1.
MME	develop	2020.w47

SPGWC	master	v1.1.0
SPGWU	master	v1.1.0

```
$ git clone https://github.com/OPENAIRINTERFACE/openair-epc-fed.git
$ cd openair-epc-fed
$ git checkout 2021.w06
$ ./scripts/syncComponents.sh --hss-branch v1.1.1 --mme-branch 2020.w47 \
                              --spgwc-branch v1.1.0 --spgwu-tiny-branch
v1.1.0
```

2. Crear las imágenes de los distintos nodos que componen el sistema

2.1. HSS

```
$ docker build --target oai-hss --tag oai-hss:production \
               --file component/oai-hss/docker/Dockerfile.ubuntu18.04 \
               component/oai-hss
$ docker image prune --force
$ docker image ls
```

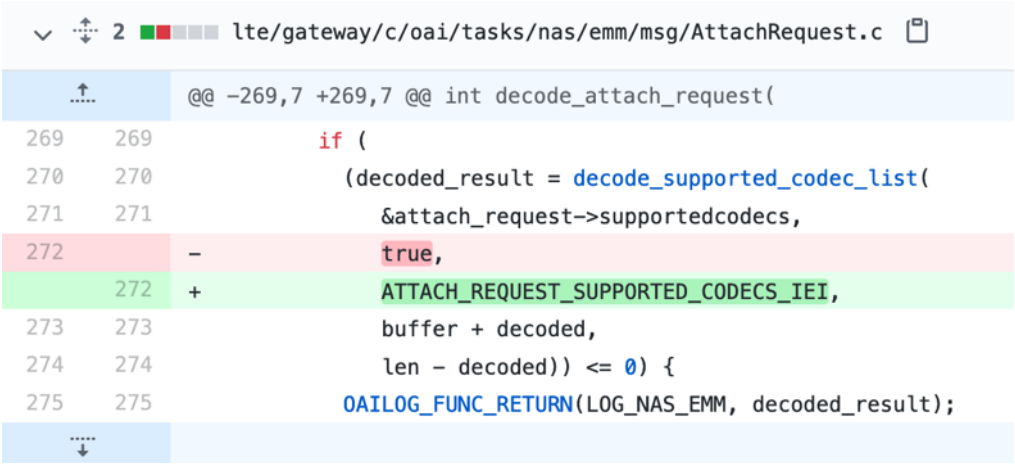
2.2. MME

```
$ docker build --target oai-mme --tag oai-mme:production \
               --file component/oai-mme/docker/Dockerfile.ubuntu18.04 \
               component/oai-mme
$ docker image prune --force
$ docker image ls
```

Una vez instalada la imagen, es importante hacer un cambio en uno de sus archivos del código fuente puesto que, si no, el UE no generaba el Attach Request y nunca se podría conectar.

```
# Go to the next directory
cd ~/openair-epc-fed/component/oai-mme/src/nas/emm/msg/

# Make a copy of the original file
sudo cp AttachRequest.c AttachReques.original.c
```

Había que realizar el siguiente cambio:



2.3. SPGWC

```
$ docker build --target oai-spgwc --tag oai-spgwc:production \
               --file component/oai-spgwc/docker/Dockerfile.ubuntu18.04 \
               component/oai-spgwc
$ docker image prune --force
$ docker image ls
```

2.4. SPGWU

```
$ docker build --target oai-spgwu-tiny --tag oai-spgwu-tiny:production \
               --file component/oai-spgwu-
tiny/docker/Dockerfile.ubuntu18.04 \
               component/oai-spgwu-tiny
$ docker image prune --force
$ docker image ls
```

En tercer lugar, **configurar de los contenedores**

1. El primer paso es crear una red virtual

```
$ sudo docker network create --attachable --subnet 192.168.61.0/26 --ip-
range 192.168.61.0/26 prod-oai-public-net
```

En esta red virtual, los nodos correspondientes tendrán las siguientes direcciones IP:

- OAI-HSS : 192.168.61.2
- OAI-MME : 192.168.61.3
- OAI-SPGWC: 192.168.61.4
- OAI-SPGWU: 192.168.61.5

2. Configurar los contenedores correspondientemente. En la configuración de los contenedores era muy importante los datos introducidos al HSS, es decir, a la "base de datos" si el objetivo es conectar un dispositivo a la red. Ejecutar los comandos dentro de la carpeta openair-epc-fed/

2.1. Cassandra

```
$ docker cp component/oai-hss/src/hss_rel14/db/oai_db.cql prod-
cassandra:/home
$ docker exec -it prod-cassandra /bin/bash -c "nodetool status"
$ Cassandra_IP=`docker inspect --format="{{range
.NetworkSettings.Networks}}{{.IPAddress}}{{end}}" prod-cassandra`
$ docker exec -it prod-cassandra /bin/bash -c "cqlsh --file
/home/oai_db.cql ${Cassandra_IP}"
```

2.2. HSS

Los datos relevantes de la tarjeta SIM a utilizar eran el OP, el IMSI y la Ki.

```
$ HSS_IP=`docker exec -it prod-oai-hss /bin/bash -c "ifconfig eth1 | grep
inet" | sed -f ./ci-scripts/convertIpAddrFromIfconfig.sed`
$ python3 component/oai-hss/ci-scripts/generateConfigFiles.py --kind=HSS --
cassandra=${Cassandra_IP} \
          --hss_s6a=${HSS_IP} --apn1=apn1.carrier.com --
apn2=apn2.carrier.com \
          --users=1 --imsi=901700000013636 \
          --ltek=68F9094FC3A0104D73823AEA4EDCBF5D --
op=5964D065DC383B2E19228E083530B71A \
          --nb_mmes=1 --from_docker_file
$ docker cp ./hss-cfg.sh prod-oai-hss:/openair-hss/scripts
$ docker exec -it prod-oai-hss /bin/bash -c "cd /openair-hss/scripts &&
chmod 777 hss-cfg.sh && ./hss-cfg.sh"
```

2.3. MME

```
$ MME_IP=`docker inspect --format="{{range
.NetworkSettings.Networks}}{{.IPAddress}}{{end}}" prod-oai-mme`
$ SPGW0_IP=`docker inspect --format="{{range
.NetworkSettings.Networks}}{{.IPAddress}}{{end}}" prod-oai-spgwc`
$ python3 component/oai-mme/ci-scripts/generateConfigFiles.py --kind=MME \
          --hss_s6a=${HSS_IP} --mme_s6a=${MME_IP} \
          --mme_s1c_IP=${MME_IP} --mme_s1c_name=eth0 \
          --mme_s10_IP=${MME_IP} --mme_s10_name=eth0 \
          --mme_s11_IP=${MME_IP} --mme_s11_name=eth0 --
spgwc0_s11_IP=${SPGW0_IP} \
          --mcc=901 --mnc=70 --tac_list="5 6 7" --from_docker_file
$ docker cp ./mme-cfg.sh prod-oai-mme:/openair-mme/scripts
$ docker exec -it prod-oai-mme /bin/bash -c "cd /openair-mme/scripts &&
chmod 777 mme-cfg.sh && ./mme-cfg.sh"
```

2.4. SPGW-C

```
$ python3 component/oai-spgwc/ci-scripts/generateConfigFiles.py --
kind=SPGW-C \
          --s11c=eth0 --sxc=eth0 --apn=apn1.carrier.com \
          --dns1_ip=YOUR_DNS_IP_ADDRESS --
dns2_ip=A_SECONDARY_DNS_IP_ADDRESS --from_docker_file
$ docker cp ./spgwc-cfg.sh prod-oai-spgwc:/openair-spgwc
$ docker exec -it prod-oai-spgwc /bin/bash -c "cd /openair-spgwc && chmod
777 spgwc-cfg.sh && ./spgwc-cfg.sh"
```

2.5. SPGW-U

```
$ python3 component/oai-spgwu-tiny/ci-scripts/generateConfigFiles.py --
kind=SPGW-U \
          --sxc_ip_addr=${SPGW0_IP} --sxu=eth0 --s1u=eth0 --
from_docker_file
$ docker cp ./spgwu-cfg.sh prod-oai-spgwu-tiny:/openair-spgwu-tiny
$ docker exec -it prod-oai-spgwu-tiny /bin/bash -c "cd /openair-spgwu-tiny
&& chmod 777 spgwu-cfg.sh && ./spgwu-cfg.sh"
```

Los contenedores ya están configurados, si se necesita modificar algún parámetro de algún nodo, es necesario eliminar los contenedores y volver a desplegarlos.

Finalmente, para poder analizar las trazas generadas entre los distintos nodos de la red, eran necesarios los siguientes comandos.

1. Para iniciar las trazas de red

```
$ docker exec -d prod-oai-hss /bin/bash -c "nohup tshark -i eth0 -i eth1 -w
/tmp/hss_check_run.pcap 2>&1 > /dev/null"

$ docker exec -d prod-oai-mme /bin/bash -c "nohup tshark -i eth0 -i lo:s10
-w /tmp/mme_check_run.pcap 2>&1 > /dev/null"

$ docker exec -d prod-oai-spgwc /bin/bash -c "nohup tshark -i eth0 -i
lo:p5c -i lo:s5c -w /tmp/spgwc_check_run.pcap 2>&1 > /dev/null"

$ docker exec -d prod-oai-spgwu-tiny /bin/bash -c "nohup tshark -i eth0 -w
/tmp/spgwu_check_run.pcap 2>&1 > /dev/null"
```

2. Para parar los contenedores

```
$ docker exec -it prod-oai-hss /bin/bash -c "killall --signal SIGINT
oai_hss tshark tcpdump"
$ docker exec -it prod-oai-mme /bin/bash -c "killall --signal SIGINT
oai_mme tshark tcpdump"
$ docker exec -it prod-oai-spgwc /bin/bash -c "killall --signal SIGINT
oai_spgwc tshark tcpdump"
$ docker exec -it prod-oai-spgwu-tiny /bin/bash -c "killall --signal SIGINT
oai_spgwu tshark tcpdump"

$ sleep 10

$ docker exec -it prod-oai-hss /bin/bash -c "killall --signal SIGKILL
oai_hss tshark tcpdump"
$ docker exec -it prod-oai-mme /bin/bash -c "killall --signal SIGKILL
oai_mme tshark tcpdump"
$ docker exec -it prod-oai-spgwc /bin/bash -c "killall --signal SIGKILL
oai_spgwc tshark tcpdump"
$ docker exec -it prod-oai-spgwu-tiny /bin/bash -c "killall --signal
SIGKILL oai_spgwu tshark tcpdump"
```

3. Para recuperar los logs, archivos de configuración y trazas en wireshark

```
$ rm -Rf archives
$ mkdir -p archives/oai-hss-cfg archives/oai-mme-cfg archives/oai-spgwc-cfg
archives/oai-spgwu-cfg
```

Recuperar los archivos de configuración

```
$ docker cp prod-oai-hss:/openair-hss/etc/. archives/oai-hss-cfg
$ docker cp prod-oai-mme:/openair-mme/etc/. archives/oai-mme-cfg
$ docker cp prod-oai-spgwc:/openair-spgwc/etc/. archives/oai-spgwc-cfg
$ docker cp prod-oai-spgwu-tiny:/openair-spgwu-tiny/etc/. archives/oai-
spgwu-cfg
```

Obtener los logs:

```
$ docker cp prod-oai-hss:/openair-hss/hss_check_run.log archives
$ docker cp prod-oai-mme:/openair-mme/mme_check_run.log archives
$ docker cp prod-oai-spgwc:/openair-spgwc/spgwc_check_run.log archives
$ docker cp prod-oai-spgwu-tiny:/openair-spgwu-tiny/spgwu_check_run.log
archives
```

Finalmente, los PCAP.

```
$ docker cp prod-oai-hss:/tmp/hss_check_run.pcap archives
$ docker cp prod-oai-mme:/tmp/mme_check_run.pcap archives
$ docker cp prod-oai-spgwc:/tmp/spgwc_check_run.pcap archives
$ docker cp prod-oai-spgwu-tiny:/tmp/spgwu_check_run.pcap archives
```

9.3.3. 5G – Core

Para la instalación del core 5G se siguió la siguiente referencia: [26]

El tag utilizado para el despliegue del core 5G fue el siguiente:
51bf03b84327b402afbf2068e1b186a4b12796ec

1. Al igual que en el EPC, en primer lugar, instalamos docker, siguiendo los mismos pasos en la **Sección 9.3.2**.
2. Para la instalación de los contenedores, existían dos opciones: descargarlos del repositorio correspondiente o configurarlos manualmente. Se escogió la primera por su sencillez.

```
$ docker login
$ docker pull rdefosseoai/oai-amf
$ docker pull rdefosseoai/oai-nrf
$ docker pull rdefosseoai/oai-spgwu-tiny
$ docker pull rdefosseoai/oai-smf
$ docker image tag rdefosseoai/oai-amf:latest oai-amf:latest
$ docker image tag rdefosseoai/oai-nrf:latest oai-nrf:latest
$ docker image tag rdefosseoai/oai-smf:latest oai-smf:latest
$ docker image tag rdefosseoai/oai-spgwu-tiny:latest oai-spgwu-tiny:latest
$ docker logout
```

3. Una vez nos hemos descargado las imágenes, clonamos el repositorio de oai 5G-CN

```
# Clone directly on the latest release tag
$ git clone --branch v1.1.0 https://gitlab.eurecom.fr/oai/cn5g/oai-cn5g-
fed.git
$ cd oai-cn5g-fed
# If you forgot to clone directly to the latest release tag
$ git checkout -f v1.1.0

# Synchronize all git submodules
$ ./scripts/syncComponents.sh
------------------------------------------------------------
OAI-NRF    component branch : master
OAI-AMF    component branch : master
OAI-SMF    component branch : master
OAI-SPGW-U component branch : master
------------------------------------------------------------
git submodule deinit --all --force
git submodule init
git submodule update
```

4. Comprobamos si es posible alcanzar los distintos nodos mediante un ping, para ello habrá que ejcutar antes de nada los siguientes comandos:

```
$ sudo sysctl net.ipv4.conf.all.forwarding=1
$ sudo iptables -P FORWARD ACCEPT
```

5. Ahora se podrán configurar los contenedores mediante el archivo **docker-compose.yaml.** Este archivo almacena información de configuración de los nodos principales. Si fuese necesario modificar algún parámetro que no se detalla en este archivo habría que configurarlo en el componente concreto, en la carpeta etc/.

En el archivo docker-compose.yaml se modificaron los siguientes parámetros:

```
1c1
< version: '3.8'
---
> version: '3.7'
50,51c50,51
<             - MCC=208
<             - MNC=95
---
>             - MCC=214
>             - MNC=07
54,55c54,55
<             - SERVED_GUAMI_MCC_0=208
<             - SERVED_GUAMI_MNC_0=95
---
>             - SERVED_GUAMI_MCC_0=214
>             - SERVED_GUAMI_MNC_0=07
62,64c62,64
<             - PLMN_SUPPORT_MCC=208
<             - PLMN_SUPPORT_MNC=95
<             - PLMN_SUPPORT_TAC=0xa000
---
>             - PLMN_SUPPORT_MCC=214
>             - PLMN_SUPPORT_MNC=07
```

```
>              - PLMN_SUPPORT_TAC=1
164,165c164,165
<              - MCC=208
<              - MNC=95
---
>              - MCC=214
>              - MNC=07
213,215d212
<     # public_net:
<     #     external:
<     #         name: demo-oai-public-net
217,223c214,223
<         driver: bridge
<         name: demo-oai-public-net
<         ipam:
<             config:
<                 - subnet: 192.168.70.128/26
<         driver_opts:
<             com.docker.network.bridge.name: "demo-oai"
---
>         external:
>             name: demo-oai-public-net
>     # public_net:
>     #     driver: bridge
>     #     name: demo-oai-public-net
>     #     ipam:
>     #         config:
>     #             - subnet: 192.168.70.128/26
>     #     driver_opts:
>     #         com.docker.network.bridge.name: "demo-oai"
```

Para poder capturar trazas del core era necesario crear la red de docker de forma manual mediante el siguiente comando:

```
(docker-compose-host)$ docker network create \
  --driver=bridge \
  --subnet=192.168.70.128/26 \
  -o "com.docker.network.bridge.name"="demo-oai" \
  demo-oai-public-net
```

Con ello, antes de ejecutar el core:

```
tshark -i demo-oai -w /tmp/5gcn-deployment.pcap
```

Para parar la ejecución del core de la red:

```
(docker-compose-host)$ sudo docker-compose -f docker-compose.yaml -p 5gcn
down
Stopping oai-ext-dn ... done
Stopping oai-amf    ... done
Stopping oai-spgwu  ... done
Stopping mysql      ... done
Stopping oai-nrf    ... done
Removing oai-ext-dn ... done
Removing oai-amf    ... done
Removing oai-smf    ... done
Removing oai-spgwu  ... done
```

```
Removing mysql      ... done
Removing oai-nrf    ... done
Network demo-oai-public-net is external, skipping
```

9.3.4. Nodos Radio

Para ambos nodos se utilizó la rama *develop,* en concreto el sha1:
f21ac1a84ca7bfa0c19e46b50b02b5078c18c427

9.3.4.1. eNB

```
$ sudo git clone https://gitlab.eurecom.fr/oai/openairinterface5g.git
$ cd openairinterface5g/
$ sudo git checkout f21ac1a84ca7bfa0c19e46b50b02b5078c18c427
$ source oaienv
$ cd cmake_targets
$ sudo ./build_oai -I -w USRP --eNB
```

9.3.4.2. gNB

```
$ sudo git clone https://gitlab.eurecom.fr/oai/openairinterface5g.git
$ cd openairinterface5g/
$ sudo git checkout f21ac1a84ca7bfa0c19e46b50b02b5078c18c427
$ source oaienv
$ cd cmake_targets
$ sudo ./build_oai -I -w USRP --gNB
```

Con esto ya se han descargado los binarios correspondientes a cada nodo:

- lte-softmodem para el eNB
- nr-softmodem para el gNB

Para ambos nodos, tanto eNB y gNB existen sendos archivos de configuración que se utilizarán al ejecutar los binarios correspondientes.

En estos archivos de configuración se pueden especificar varios parámetros de los nodos, entre los cuales se encuentran los siguientes:

- MCC, MNC y TAC, al especificar estos tres valores es importante que concuerden tanto en el archivo de configuración del nodo eNB como del gNB.
- Direcciones IP
  - MME: 192.168.61.3
  - eNB: 192.168.1.88
  - gNB: 192.168.1.91
- Parámetros físicos, es decir, los valores relacionados con las frecuencias tanto de subida como de bajada, el ancho de banda, el modo de transmisión (TDD/FDD), etc.
- Si utilizar referencia de reloj externa, aunque es necesario tener un GPS.

Debido a la extensión del archive de configuración, solo se van a mostrar los cambios realizados.

eNB – Archivo de configuración

El archivo completo se puede encontrar:

- En la siguiente url: https://gitlab.eurecom.fr/oai/openairinterface5g/-/blob/develop/ci-scripts/conf_files/enb.band7.tm1.fr1.25PRB.usrpb210.conf
- Dentro de la siguiente ruta también aparecen muchos archivos de configuración a utilizar: ***openairinterface5g/targets/PROJECTS/GENERIC-LTE-EPC/CONF***

Los cambios realizados fueron los siguientes:

```
18c18
<       { mcc = 222; mnc = 01; mnc_length = 2; }
---
>       { mcc = 214; mnc = 07; mnc_length = 2; }
184c184
<     mme_ip_address      = ( { ipv4       = "CI_MME_IP_ADDR";
---
>     mme_ip_address      = ( { ipv4       = "192.168.61.3";
201,202c201,202
<         ENB_INTERFACE_NAME_FOR_S1_MME            = "eth1";
<         ENB_IPV4_ADDRESS_FOR_S1_MME              = "CI_ENB_IP_ADDR";
---
>         ENB_INTERFACE_NAME_FOR_S1_MME            = "eth0";
>         ENB_IPV4_ADDRESS_FOR_S1_MME              = "192.168.1.88";
204,205c204,205
<         ENB_INTERFACE_NAME_FOR_S1U               = "eth1";
<         ENB_IPV4_ADDRESS_FOR_S1U                 = "CI_ENB_IP_ADDR";
---
>         ENB_INTERFACE_NAME_FOR_S1U               = "eth0";
>         ENB_IPV4_ADDRESS_FOR_S1U                 = "192.168.1.88";
208c208
<         ENB_IPV4_ADDRESS_FOR_X2C                 = "CI_ENB_IP_ADDR";
---
>         ENB_IPV4_ADDRESS_FOR_X2C                 = "192.168.1.88";
247c247
<     worker_config    = "WORKER_ENABLE";
---
>     worker_config    = "ENABLE";
263c263
<     att_tx          = 3
---
>     att_tx          = 0
267c267
<     max_rxgain                    = 115;
---
>     max_rxgain                    = 118;
289a290
>
```

gNB – Archivo de Configuración

Al igual que en el nodo eNB se pueden encontrar los archivos de configuración en dos partes:

- En la siguiente url: https://gitlab.eurecom.fr/oai/openairinterface5g/-/blob/develop/ci-scripts/conf_files/gnb.band78.tm1.fr1.106PRB.usrpb210.conf
- Dentro de la siguiente ruta: ***openairinterface5g/targets/PROJECTS/GENERIC-LTE-EPC/CONF***

```
3c3
< Asn1_verbosity = "none";
---
> Asn1_verbosity = "annoying";
17c17
<     plmn_list = ({mcc = 901; mnc = 70; mnc_length = 2;});
---
>     plmn_list = ({mcc = 214; mnc = 07; mnc_length = 2;});
232c232
<         GNB_IPV4_ADDRESS_FOR_S1_MME              = "192.168.1.90";
---
>         GNB_IPV4_ADDRESS_FOR_S1_MME              = "192.168.1.91";
234c234
<         GNB_IPV4_ADDRESS_FOR_S1U                 = "192.168.1.90";
---
>         GNB_IPV4_ADDRESS_FOR_S1U                 = "192.168.1.91";
236,237c236,237
<         GNB_IPV4_ADDRESS_FOR_X2C                 = "192.168.1.90";
<         GNB_PORT_FOR_X2C                         = 36422; # Spec 36422
---
>         GNB_IPV4_ADDRESS_FOR_X2C                 = "192.168.1.91";
>         GNB_PORT_FOR_X2C                         = 36422 # Spec 36422
```

9.3.5. Programación de las tarjetas SIM

```
$ git clone https://github.com/herlesupreeth/sysmo-usim-tool
```

Al descargar el repositorio habrá varios scripts de Python que se podrán ejecutar para programar las tarjetas SIM. Es importante aclarar que se utiliza la versión de Python 2.7.

Además, son necesarias varias dependencias:

- Pyscard

```
$ pip install pyscard
```

También es posible que se necesite instalar pcscd service

```
miguel@miguel-NUC10i7FNH:~/Documents/OAI/progrom_sim_cards/sysmo-usim-tool$
./sysmo-isim-tool.sja2.py --help

sysmoISIM-SJA2 parameterization tool
Copyright (c)2019 Sysmocom s.f.m.c. GmbH

 * Commandline options:
   -h, --help ..................... Show this screen
   -f, --force .................... Enforce authentication after failure
   -a, --adm1 CHV ................. Administrator PIN (e.g 55538407)
   -J, --set-imsi ................. Set IMSI value
   -n, --mnclen ................... Show MNC length value
   -N, --set-mnclen ............... Set MNC length value
   -l, --milenage ................. Show milenage parameters
   -L, --set-milenage HEXSTRING ... Set milenage parameters
   -k, --ki ....................... Show KI value
   -K, --set-ki ................... Set KI value
   -t, --auth ..................... Show Authentication algorithms
   -T, --set-auth 2G:3G ........... Set 2G/3G Auth algo (e.g.
COMP128v1:COMP128v1)
   -o, --opc ...................... Show OP/c configuration
   -O, --set-op HEXSTRING ......... Set OP value
   -C, --set-opc HEXSTRING ........ Set OPc value
   -s  --seq-parameters ........... Show MILENAGE SEQ/SQN parameters
   -S  --reset-seq-parameters ..... Reset MILENAGE SEQ/SQN parameters to
default
   -i  --iccid .................... Show ICCID
   -p  --aid ...................... Show AID list (installed applications)
   -d, --dump ..................... Dump propritary file contents

   For Option -T, the following algorithms are valid:
   1 COMP128v1
   2 COMP128v2
   3 XOR-2G
   4 MILENAGE
   5 SHA1-AKA
   15 XOR
```

9.4. Mensajes Intercambiados

9.4.1. Mensaje Inicial del gNB al eNB

Para que los nodos radio se puedan comunicar entre ellos, el gNB enviará un mensaje mediante el protocolo X2AP. Este protocolo se utiliza para procedimientos elementares entre los nodos radio, mediante la interfaz X2. Como se puede comprobar en el siguiente código, el mensaje es un SetupRequest incluyendo información como el ID del nodo o la identidad de su PLMN.

```
<X2AP-PDU>
    <initiatingMessage>
        <procedureCode>36</procedureCode>
        <criticality><reject/></criticality>
        <value>
            <ENDCX2SetupRequest>
                <protocolIEs>
                    <ENDCX2SetupRequest-IEs>
```

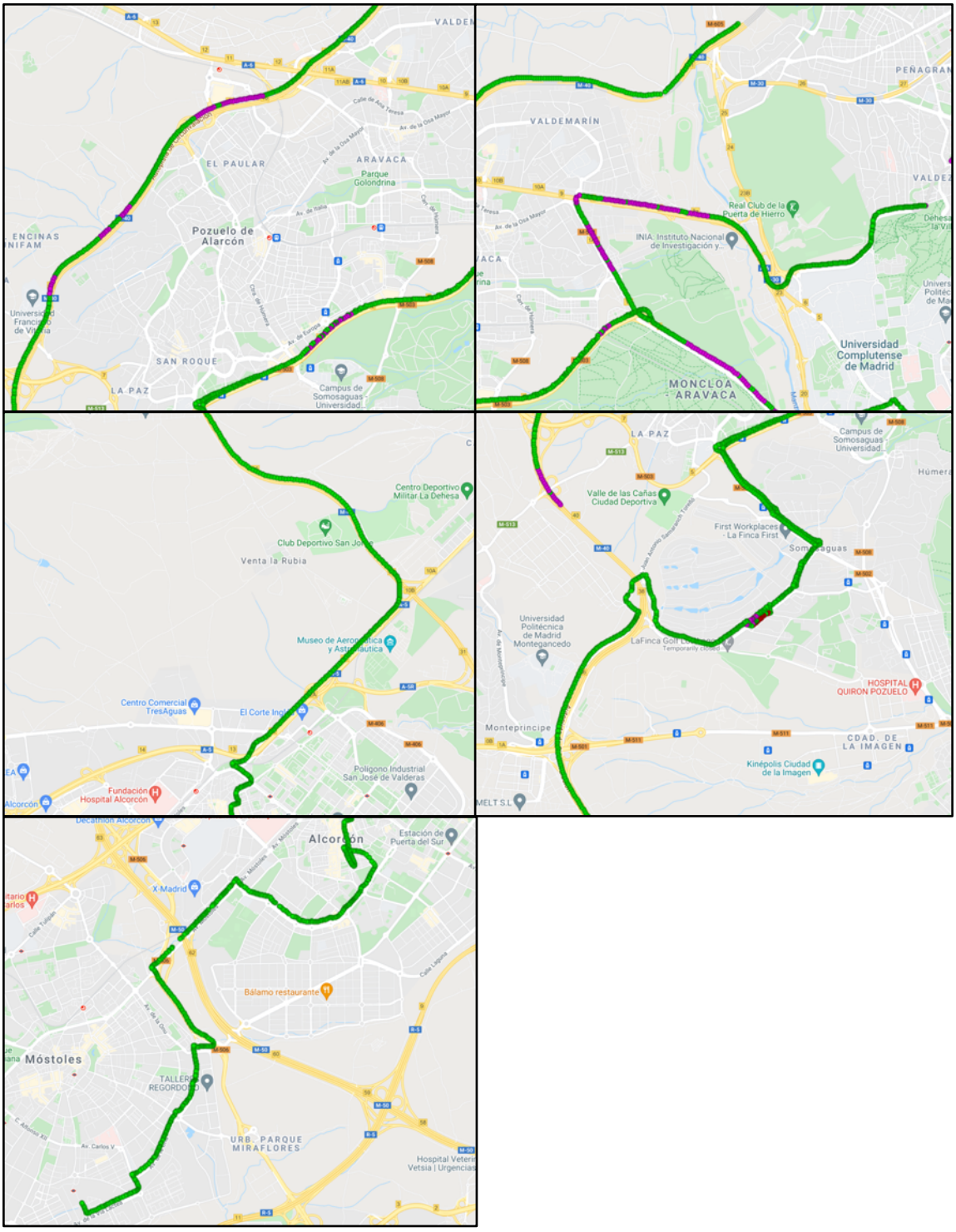
```
                        <id>244</id>
                        <criticality><reject/></criticality>
                        <value>
                            <InitiatingNodeType-EndcX2Setup>
                                <init-en-gNB>
                                    <En-gNB-ENDCX2SetupReqIEs>
                                        <id>252</id>

<criticality><reject/></criticality>
                                        <value>
                                            <GlobalGNB-ID>
                                                <pLMN-Identity>12 F4
70</pLMN-Identity>
                                                <gNB-ID>
                                                    <gNB-ID>

00000000000000000000111000000000
                                                    </gNB-ID>
                                                </gNB-ID>
                                            </GlobalGNB-ID>
                                        </value>
                                    </En-gNB-ENDCX2SetupReqIEs>
                                    <En-gNB-ENDCX2SetupReqIEs>
                                        <id>253</id>

<criticality><reject/></criticality>
                                        <value>

<ServedNRcellsENDCX2ManagementList>
                                                <SEQUENCE>
                                                    <servedNRCellInfo>
                                                        <nrpCI>0</nrpCI>
                                                        <nrCellID>
                                                            <pLMN-
Identity>12 F4 70</pLMN-Identity>

<nRcellIdentifier>

000000000000000000000000111000000000

</nRcellIdentifier>
                                                        </nrCellID>
                                                        <fiveGS-TAC>00 00
01</fiveGS-TAC>
                                                        <broadcastPLMNs>
                                                            <PLMN-
Identity>12 F4 70</PLMN-Identity>
                                                        </broadcastPLMNs>
                                                        <nrModeInfo>
                                                            <tdd>


<nRFreqInfo>

<nRARFCN>640000</nRARFCN>

<freqBandListNr>

<FreqBandNrItem>

<freqBandIndicatorNr>78</freqBandIndicatorNr>
```

```
<supportedSULBandList>

</supportedSULBandList>

</FreqBandNrItem>

</freqBandListNr>

</nRFreqInfo>
                                                                        <nR-TxBW>

<nRSCS><scs15/></nRSCS>

<nRNRB><nrb106/></nRNRB>
                                                                        </nR-TxBW>
                                                                    </tdd>
                                                                </nrModeInfo>

<measurementTimingConfiguration>00</measurementTimingConfiguration>
                                                                    </servedNRCellInfo>
                                                            </SEQUENCE>

</ServedNRcellsENDCX2ManagementList>
                                            </value>
                                        </En-gNB-ENDCX2SetupReqIEs>
                                    </init-en-gNB>
                                </InitiatingNodeType-EndcX2Setup>
                            </value>
                        </ENDCX2SetupRequest-IEs>
                    </protocolIEs>
                </ENDCX2SetupRequest>
            </value>
        </initiatingMessage>
</X2AP-PDU>
```

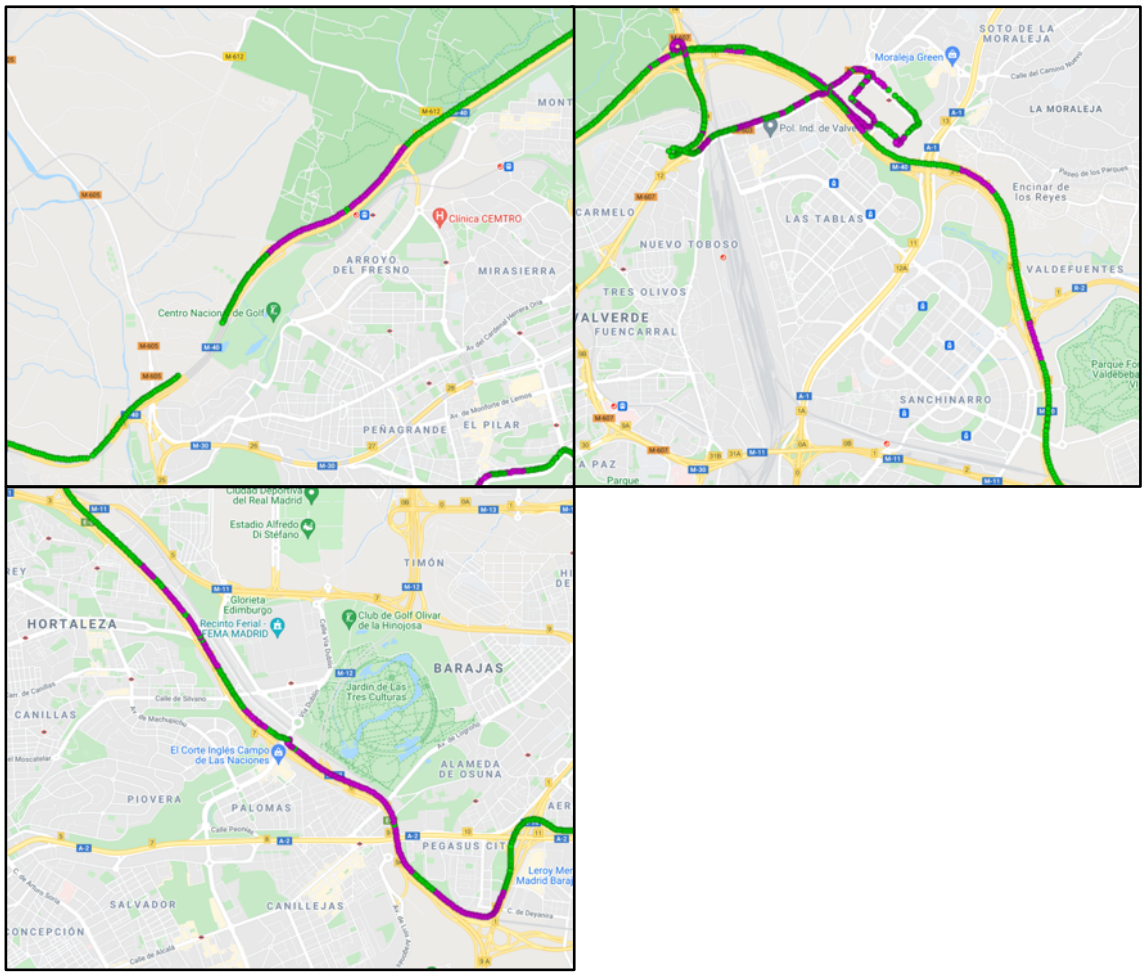
9.5. Wardriving

En las imágenes siguientes el color representa la tecnología a la que estaba el dispositivo móvil conectado, siendo de color verde 4G-LTE y de color morado 5G, tanto NSA como SA.

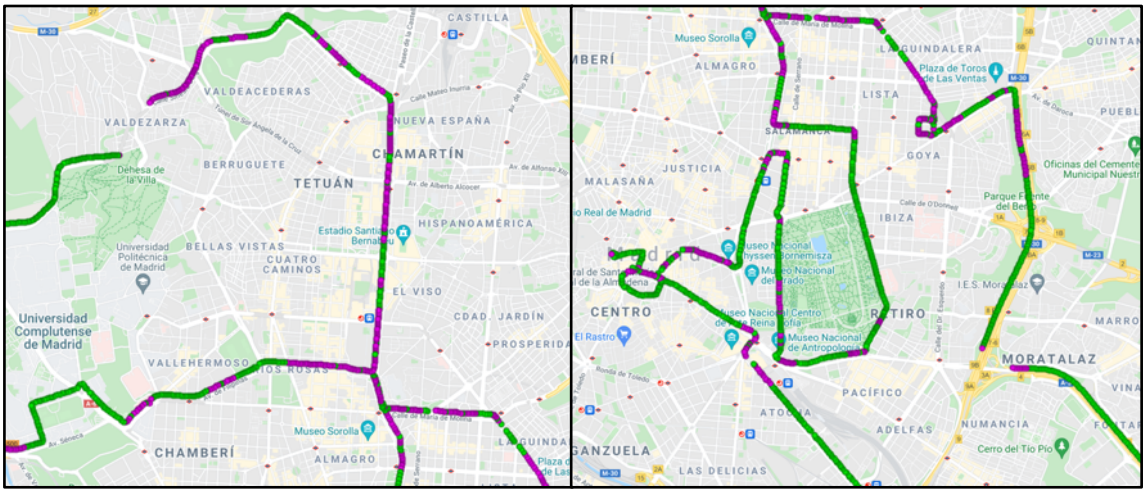
9.5.1. ZONA OESTE



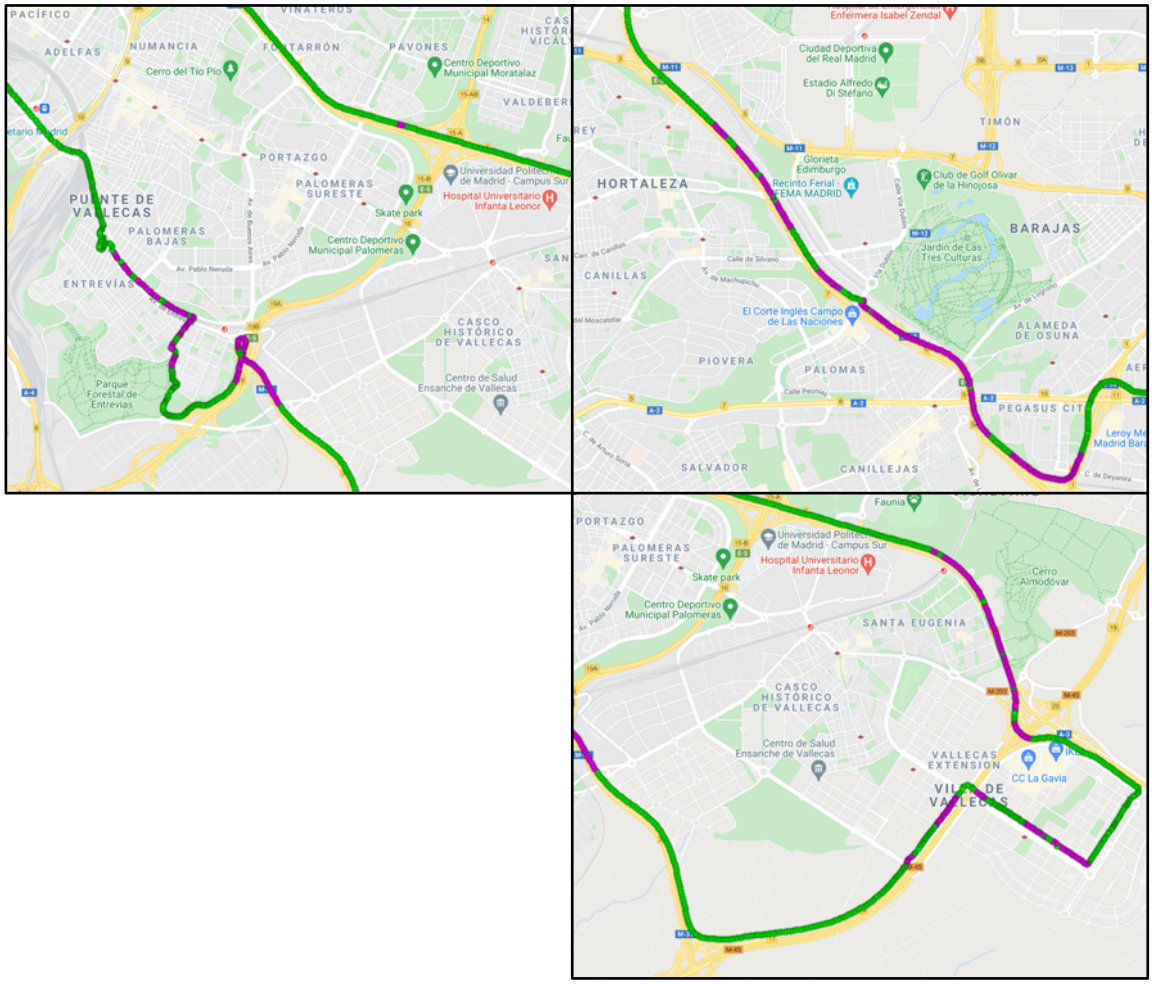
9.5.2. ZONA NORTE



9.5.3. ZONA CENTRO



9.5.4. ZONA SUR


PACÍFICO
ADELFAS
NUMANCIA
Cerro del Tío Pío
PORTAZGO
PALOMERAS SURESTE
Skate park
Universidad Politécnica de Madrid - Campus Sur
Hospital Universitario Infanta Leonor
Centro Deportivo Municipal Palomeras
PUENTE DE VALLECAS
PALOMERAS BAJAS
ENTREVÍAS
Parque Forestal de Entrevías
CASCO HISTÓRICO DE VALLECAS
Centro de Salud Ensanche de Vallecas
PAVONES
Centro Deportivo Municipal Moratalaz
VALDEBER
Enfermera Isabel Zendal
Ciudad Deportiva del Real Madrid
Estadio Alfredo Di Stéfano
TIMÓN
HORTALEZA
Glorieta Edimburgo
Recinto Ferial - IFEMA MADRID
Club de Golf Olivar de la Hinojosa
BARAJAS
Jardín de Las Tres Culturas
CANILLAS
El Corte Inglés Campo de Las Naciones
ALAMEDA DE OSUNA
PIOVERA
PALOMAS
PEGASUS CITY
SALVADOR
CANILLEJAS
Faunia
Cerro Almodóvar
SANTA EUGENIA
VALLECAS EXTENSIÓN
CC La Gavia