

TD / TP appliqué au réseau 5G (Autour de free5gc/docker) Master2 RSA

Suzanne Chan (Déc. 2023/ Janv. 2024)

Introduction

Notre présente étude autour de la virtualisation, le cloud appliquée à l'architecture 5G nous conduit à sélectionner le projet open source free5gc stage3 docker, afin de permettre aux étudiants du master2 RSA d'appréhender rapidement et de développer les acquis dans le vaste monde de la 5G.

Cela leur permet d'aborder par la pratique les multiples facettes de cette nouvelle génération du réseau mobile. Ils pourront porter un regard transverse sur les aspects IT, réseau, protocole 5G et sécurité.

Afin de faciliter ce développement, cette présentation est accompagnée d'un mode opératoire détaillé pour aider dans l'installation de free5gc en mode docker dans un environnement Linux, Ubuntu de préférence.

Etude : Autour de free5GC et trame

Environnements

- Linux
 - Environnement
 - Système
- Protocoles réseaux
 - Model OSI
- Prérequis
 - GIT
 - Docker

Objectif

- Installer free5GC stage 3 / UEransim
- Utilisation / Etudes traces
 - Configuration du 5GC
 - Docker

Evaluation

- Evaluation expliquée en slide n°7

Objet de l'étude

L'objectif des travaux pour l'évaluation est

1. de configurer son propre réseau et non plus celui donné par défaut avec l'installation.
2. de pouvoir utiliser les commandes docker
3. de pouvoir construire plusieurs instances ueransim pour établir des échanges sur la "Data plane" vu lors des séances en présentiel
4. de capturer ses propres traces wireshark générées par ces échanges pour l'étude protocolaire et d'observer ce tunnel gtp-u.

Les filtres wireshark sont (SBI : pour Service Based Interface)

N2 : ngap

SBI : http2

N4 : pfcf

Rappel/synthèse des notions abordées en séances 1/2:

Histoire autour des procédures 5G d'activation du mobile depuis le registration request à la construction du tunnel pdu gtp-u dans le user plane, en passant par les procédures de sécurisation et d'authentification de l'UE (User Equipment) identifié par son IMSI...

Pour un abonnement donné, l'opérateur free5gc enregistre l'IMSI paramétrage dans la base de donnée MongoDB... Les détails du profil de l'abonné, pour chaque paramètre ont été présentés. Parmi eux se trouvent les clés K et OPC de sécurité, les slices, l'apn ou dnn en 5G...

Pour un abonnement donné, l'opérateur pourra appliquer des règles de filtrages et QoS (qualité de service) à travers la fonction PCF, SMF et UPF (pour Policy and Charging, Session Management et User Plane Functions) via le protocole PFCP et l'interface N4.

Autour du system linux, le choix du docker permet de proposer un environnement de virtualisation très léger pour gérer les infrastructures réseaux telle la gestion des applications logicielles avec rapidité et facilité. Notre étude porte sur les commandes proposées par docker et les applications des NF (network function) 5G proposées par le projet open source.

Rappel/synthèse des notions abordées en séances 2/2:

Par le biais de docker, nous déployons le système applicatif free5gc et ueransim conteneurisé à partir de la configuration de base de manière à pouvoir multiplier le nombre de UE et de NR 5G ou gNB (pour New Radio respectivement gNodeB).

Nous avons également abordé l'aspect réseau avec la création d'une seconde interface et d'un deuxième sous réseau dans l'UPF pour insister dans la séparation éventuelle des interfaces N3 et N6.

L'objectif de cette étude est de permettre à l'étudiant d'avoir les outils nécessaires pour poursuivre dans les travaux autour de ce déploiement d'infrastructure applicative 5G. La formation complémentaire théorique lui permet de réagir et d'identifier plus facilement les cas de panne généralement liés à des problématiques de défaut de configurations. L'analyse du protocole 5G est un des moyens de résolution.

L'étudiant est incité à créer lui-même certaines pannes réseaux telles que :

- Stopper l'UPF pour observer les réactions
- Activer un abonné dont la base n'est pas provisionnée
- Activer un abonné avec un « requested slice » non valide
- ...

Pour l'évaluation à la fin de nos séances de travaux:

Il est demandé à l'étudiant de réaliser (seul ou en binôme) un **rapport de synthèse** de ses **propres vision et développement suite à cette phase d'étude free5gc**, en exposant les problèmes rencontrés (**par exemple à l'installation, au déploiement etc...**) Ainsi que les cas d'usages demandés définis dans le slide n°8 afin de compléter sa capacité à concevoir.

Dans cette optique, je vous accompagne par ce présent document qui résume les séances passées. Le slide 11 présente les manipulations concrètes réalisées durant le 10 et 11 janvier dernier. Afin d'évaluer concrètement l'esprit de synthèse pour cette étude ainsi que la compréhension globale des notions acquises, il est demandé aux étudiants de finir la partie (non abordée en séances) « Configurer le routage » pour le cas d'usage suivant:

Une application Internet sur protocole UDP, nommée UDPapp serveur (voix sur IP par exemple sur la partie N6), échange avec une application cliente nommée UDPapp client sur un mobile.

Cela se concrétisera par l'établissement d'un schéma de routage entre ces deux applications dans le user plane (UP) schéma slide 10 entre le mobile/gNB (ueransim) et l'UPF ainsi que la partie N6.

Un petit rappel des notions de routage se trouve dans le slide 13 pour rappel.

Use case (Devoir à la maison janv. 2024)

Le cas d'usage est le suivant :

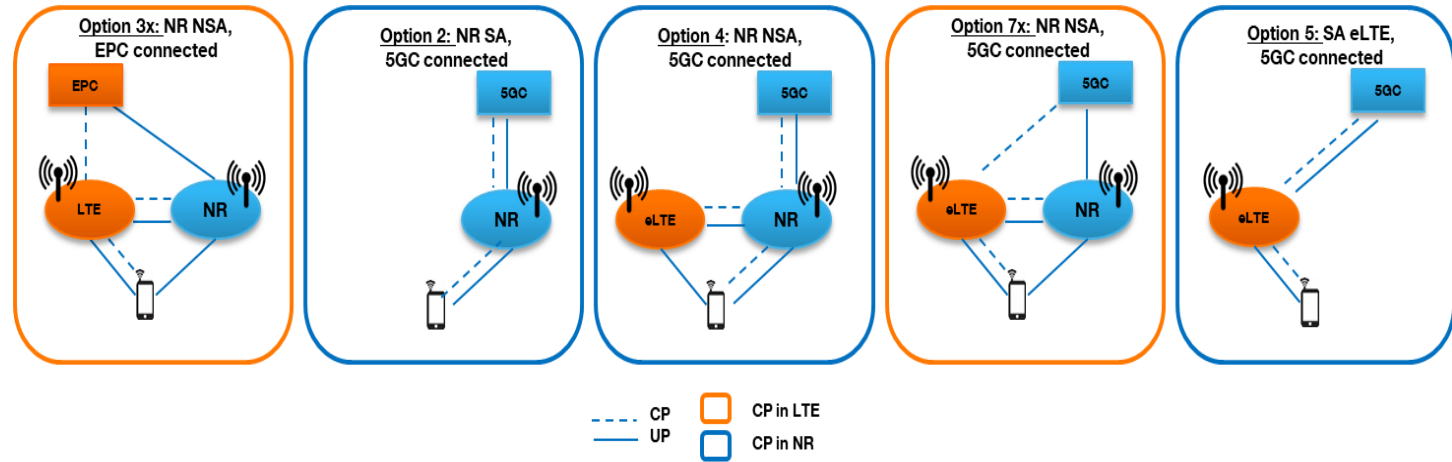
Une application Internet sur protocole UDP, nommée UDPapp serveur (voix sur IP par exemple sur la partie N6), échange avec une application cliente nommée UDPapp client sur un mobile.

1. Réaliser un schéma applicatif décrivant une situation de HO (Hand Over) dans lequel le mobile bouge d'un gNB à un autre...
 - METTRE EN EVIDENCE LES DEUX TUNNELS GTP-U
 - Le mode concerné est SSC1: une seule adresse IP (break before make)
2. Réaliser un schéma applicatif décrivant la route empruntée par :
 - un paquet descendant parcourant vers N6 UPF...
 - un paquet montant de l'application dans le mobile vers l'interface N3 gtpu (GNB) ...
 - METTRE EN EVIDENCE LES TUNNELS GTP-U et « uesimtunX », LES ROUTES au niveau GNB_HOST... ET UPF.
3. Étude de faisabilité et mise en application avec la commande « `ip route add IPdestination via GW` »
 - Attention ici l'application nr-ue de ueransim génère un tunnel « uesimtunX », attention donc aux adresses ip martiennes.

L'objectif de ce cas d'usage est de mettre en évidence l'observation du tunnel gtp-u d'où le travail sur les règles et la configuration des différentes routes en sortant le gNB ueransim dans la machine HOST... Car le conteneur ne contient que le strict minimum pour les besoins de services. Pour l'observation, nous avons besoin des outils tels que tshark non disponible dans le container.

Autour de free5GC

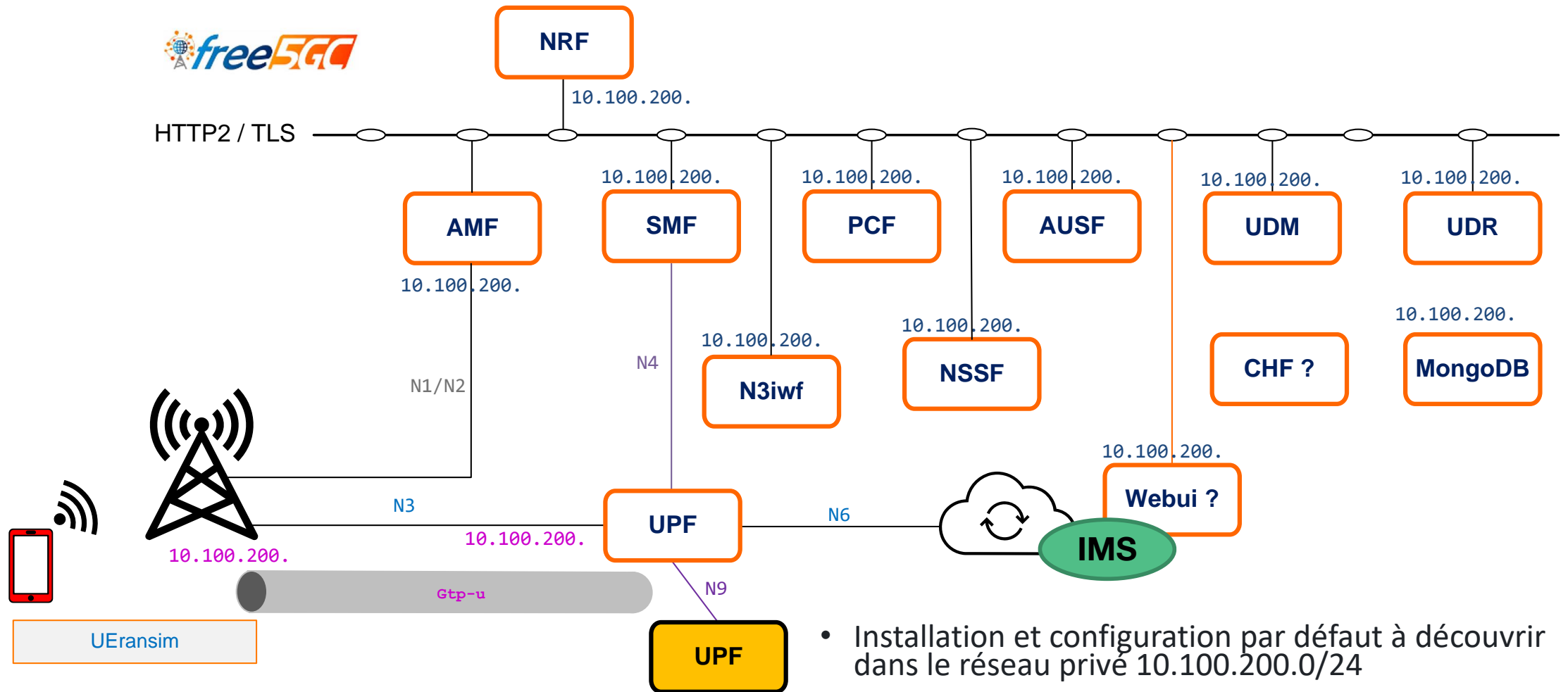
- Stage1 (Option5)
 - C
- Stage2 (Option2)
 - Go
 - UPF (C)
- Stage3 (Docker)



TP : Stage 3

- Installation
 - Docker
 - <https://github.com/free5gc/free5gc-compose>
- Qu'est-ce que le projet free5GC
 - [free5GC](https://github.com/free5gc/free5gc)
 - [GitHub - free5gc/free5gc: Open source 5G core network base on 3GPP R15](https://github.com/free5gc/free5gc)
- A propos de docker
 - <https://docs.docker.com/get-started/overview/>

Architecture stage3 (Déc. 2023)



- Installation et configuration par défaut à découvrir dans le réseau privé 10.100.200.0/24

Manip docker (Janvier 2024)

- Multiplier les UE ☒
- Multiplier le gNB ☒
 - Conteneur ueransim dans la machine host ...
- Multiplier le network
 - Conteneur docker UPF :
 - Ajouter un network ... ☒
 - Sortir le gNB dans le host (à réaliser devoir à la maison)
 - Configurer le routage ☒
 - Capture tshark ☒ ☒ / lecture dans widows, filtre http2 nok sous linux (version wireshark non compatible)
- Manip traces et protocoles :
 - Scenarii ... et validation du free5gc
 - Découverte et commandes ./nr-ue pour les changements d'états IDLE / CONNECTED
 - Stopper l'UPF et observer ☒
 - Connecter sur un slice inexistant et observer
 - ...
- Définir des architectures réseaux pour le déploiement des infrastructures applicatives free5gc
 - Mettre en application
 - Etudes de faisabilité et construction

Manip Docker / Éléments clés (Janvier 2024)

- Découvrir le pool d'adressage attribué aux UE ✓
- Etablir la connexion d'un UE new radio 5G dans le conteneur ueransim ✓
 - Etude de la configuration de création d'un UE
 - Dupliquer la configuration de l'UE en identifiant ce qui le caractérise
 - Multiplier selon besoin ...
- Etablir des sessions NGSetup d'un GNB new radio 5G avec le conteneur ueransim ✓
 - Etude de la configuration de création d'un gNodeB new radio
 - Dupliquer la configuration du conteneur ueransim en identifiant ce qui le caractérise
 - Multiplier les gNB selon besoin avec les commandes docker compose ...
- Possibilité de sortir le conteneur ueransim dans la machine host ... ✓
 - Option réseau en `network_mode : host`
- Multiplier le network / dans l'UPF par exemple en séparant N3 et N6 dans 2 sous réseaux
 - Etude de la configuration de création du réseau dans le docker compose file
 - Séparer les 2 blocs « privnet » ...
 - Analyser la configuration iptables si nécessaire
 - Ajouter la seconde interface dans l'UPF un network ... ✓

Éléments clés / routages (Janvier 2024)

- Le model OSI définit le niveau routage afin de délivrer un paquet IP d'une source vers une destination:
- Le paquet IP est caractérisé par
 - IP source, port source | IP destination, port destination | Entête protocole de transport | DATA...
 - Le protocole de transport peut être : UDP, TCP ou SCTP.
- Le routeur agit uniquement au niveau IP, il analyse en général uniquement l'adresse de destination à des exceptions près de règles de filtrages ...
pour une IP de destination, on le route vers une GateWay donnée
- Des exemples de règles de routage sont donnés par la commande « ip route »
default via 10.10.22.1 dev enp8s0f0 onlink
10.100.200.0/24 dev br-free5gc proto kernel scope link src 10.100.200.1
10.100.201.0/24 dev br2-free5gc proto kernel scope link src 10.100.201.1
10.100.202.0/24 via 10.100.200.3 dev br-free5gc
10.10.22.0/27 dev enp8s0f0 proto kernel scope link src 10.10.22.7
10.17.0.0/16 dev docker0 proto kernel scope link src 10.17.0.1 linkdown
10.20.0.0/16 dev br-d9649e712199 proto kernel scope link src 10.20.0.1

Manip docker / Conteneur ueransim

[Host:] Information Container

```
docker ps
docker network ls
docker inspect free5gc-compose_privnet
```

```
docker exec -it ueransim bash
root@10dbfe99162c:/ueransim#
```

[ueransim:] gNB est lancé automatiquement lors de la création

```
# docker restart ueransim
```

⇒ Trace SCTP ... Protocole NGAP / Procédure de “NGSetup Request”

```
20 17.892923094 10.100.200.17 → 10.100.200.15 SCTP 82 INIT
21 17.893036716 10.100.200.15 → 10.100.200.17 SCTP 306 INIT_ACK
22 17.893084268 10.100.200.17 → 10.100.200.15 SCTP 278 COOKIE_ECHO
23 17.893137556 10.100.200.15 → 10.100.200.17 SCTP 50 COOKIE_ACK
24 17.893922986 10.100.200.17 → 10.100.200.15 NGAP 134 NGSetupRequest
25 17.893963772 10.100.200.15 → 10.100.200.17 SCTP 62 SACK (Ack=0, Arwnd=106424)
26 17.900063835 10.100.200.15 → 10.100.200.17 NGAP 118 NGSetupResponse
27 17.900153437 10.100.200.17 → 10.100.200.15 SCTP 62 SACK (Ack=0, Arwnd=106443)
28 17.984024339 10.100.200.15 → 10.0.0.1      SCTP 106 HEARTBEAT
```

Manip docker / Conteneur ueransim

[ueransim:] nr-ue pour New radio (5G) User Equipment

```
# docker exec -it ueransim bash
root@10dbfe99162c:/ueransim# ./nr-ue -c config/uecfg.yaml
```

⇒ Trace SCTP ... Protocole NGAP / Procédure de “Registration Request” & PDU session Establishment Request”

```
61 51.936000736 10.100.200.15 → 10.0.0.1      SCTP 106 HEARTBEAT
62 53.151990823 10.100.200.15 → 10.0.0.1      SCTP 106 HEARTBEAT
63 54.080150841 10.100.200.17 → 10.100.200.15 NGAP/NAS-5GS 138 InitialUEMessage, Registration request
64 54.121807879 10.100.200.15 → 10.100.200.17 NGAP/NAS-5GS 146 SACK (Ack=1, Arwnd=106496) , DownlinkNASTransport, Authentication request
65 54.123709537 10.100.200.17 → 10.100.200.15 NGAP/NAS-5GS 146 SACK (Ack=1, Arwnd=106496) , UplinkNASTransport, Authentication response
66 54.138808122 10.100.200.15 → 10.100.200.17 NGAP/NAS-5GS 126 SACK (Ack=2, Arwnd=106496) , DownlinkNASTransport, Security mode command
67 54.140444349 10.100.200.17 → 10.100.200.15 NGAP/NAS-5GS 190 SACK (Ack=2, Arwnd=106496) , UplinkNASTransport
68 54.208594071 10.100.200.15 → 10.100.200.17 NGAP/NAS-5GS 246 SACK (Ack=3, Arwnd=106496) , InitialContextSetupRequest
69 54.209165647 10.100.200.17 → 10.100.200.15 NGAP 98 SACK (Ack=3, Arwnd=106496) , InitialContextSetupResponse
70 54.411988678 10.100.200.15 → 10.100.200.17 SCTP 62 SACK (Ack=4, Arwnd=106496)
71 54.412063309 10.100.200.17 → 10.100.200.15 NGAP/NAS-5GS 242 UplinkNASTransport, UplinkNASTransport
72 54.505418890 10.100.200.15 → 10.100.200.17 NGAP/NAS-5GS 258 SACK (Ack=6, Arwnd=106496) , PDUSessionResourceSetupRequest
73 54.512298900 10.100.200.17 → 10.100.200.15 NGAP 118 SACK (Ack=4, Arwnd=106496) , PDUSessionResourceSetupResponse
74 54.715988678 10.100.200.15 → 10.100.200.17 SCTP 62 SACK (Ack=7, Arwnd=106496)
75 54.848002319 10.100.200.15 → 10.0.0.1      SCTP 106 HEARTBEAT
```

Ueransim / utilisation & découverte

```
@8a8276a8dbd7:/ueransim# ./nr-cli --dump
UERANSIM-gnb-208-93-1
imsi-208930000000003
```

```
@8a8276a8dbd7:/ueransim# ./nr-cli imsi-208930000000003
```

```
-----
$ commands
info          | Show some information about the UE
status        | Show some status information about the UE
timers        | Dump current status of the timers in the UE
rls-state     | Show status information about RLS
coverage      | Dump available cells and PLMNs in the coverage
ps-establish  | Trigger a PDU session establishment procedure
ps-list       | List all PDU sessions
ps-release    | Trigger a PDU session release procedure
ps-release-all | Trigger PDU session release procedures for all active sessions
deregister    | Perform a de-registration by the UE
-----
```

```
$ info
supi: imsi-208930000000003
hplmn: 208/93
imei: 356938035643803
imeisv: 4370816125816151
ecall-only: false
uac-aic:
  mps: false
  mcs: false
uac-acc:
  normal-class: 0
  class-11: false
  class-12: false
  class-13: false
  class-14: false
  class-15: false
is-high-priority: false
```


Ueransim / utilisation

```
# docker exec -it ueransim bash
```

```
root@6902f12f8138:/ueransim# ip a
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: ip_vti0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN group default qlen 1000
    link/ipip 0.0.0.0 brd 0.0.0.0
3: uesimtun0: <POINTOPOINT,PROMISC,NOTRAILERS,UP,LOWER_UP> mtu 1400 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 10.60.0.9/32 scope global uesimtun0
        valid_lft forever preferred_lft forever
4: uesimtun1: <POINTOPOINT,PROMISC,NOTRAILERS,UP,LOWER_UP> mtu 1400 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 10.60.0.10/32 scope global uesimtun1
        valid_lft forever preferred_lft forever
534: eth0@if535: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:64:c8:11 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.100.200.17/24 brd 10.100.200.255 scope global eth0
        valid_lft forever preferred_lft forever
```

```
root@6902f12f8138:/ueransim# ping 10.60.0.9
```

```
PING 10.60.0.9 (10.60.0.9) 56(84) bytes of data.
64 bytes from 10.60.0.9: icmp_seq=1 ttl=64 time=0.026 ms
64 bytes from 10.60.0.9: icmp_seq=2 ttl=64 time=0.023 ms
^F^C
--- 10.60.0.9 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1027ms
rtt min/avg/max/mdev = 0.023/0.024/0.026/0.001 ms
```

```
root@6902f12f8138:/ueransim# ping 10.60.0.10
```

```
PING 10.60.0.10 (10.60.0.10) 56(84) bytes of data.
64 bytes from 10.60.0.10: icmp_seq=1 ttl=64 time=0.023 ms
64 bytes from 10.60.0.10: icmp_seq=2 ttl=64 time=0.020 ms
^C^F
--- 10.60.0.10 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.020/0.021/0.023/0.001 ms
```

```
root@6902f12f8138:/ueransim# ping 10.60.0.10 -I
10.60.0.9
```

```
PING 10.60.0.10 (10.60.0.10) from 10.60.0.9 : 56(84) bytes of data.
64 bytes from 10.60.0.10: icmp_seq=1 ttl=64 time=0.034 ms
64 bytes from 10.60.0.10: icmp_seq=2 ttl=64 time=0.020 ms
64 bytes from 10.60.0.10: icmp_seq=3 ttl=64 time=0.020 ms
[...]
64 bytes from 10.60.0.10: icmp_seq=22 ttl=64 time=0.020 ms
64 bytes from 10.60.0.10: icmp_seq=23 ttl=64 time=0.020 ms
64 bytes from 10.60.0.10: icmp_seq=24 ttl=64 time=0.021 ms
^C
--- 10.60.0.10 ping statistics ---
24 packets transmitted, 24 received, 0% packet loss, time 23535ms
rtt min/avg/max/mdev = 0.017/0.021/0.034/0.003 ms
```

ANNEXE & résumé vu en séances

Master2 RSA

Suzanne Chan (Déc. 2023/ Janv. 2024)

Free5gc Stage3 / Docker

Master2 RSA

Suzanne Chan (Déc. 2023/ Janv. 2024)

Free5gc Stage3 / Docker

Installation ✓

Découverte : Réseau ✓ / Schéma (objectif) ✓

Moyens

Instructions docker ✓

System et environnement linux ✓

Wireshark (tshark -i br-free5gc -w /tmp/trace.pcap) ✓



free5gcG

Prérequis

Linux Ubuntu (préférable)

git

Docker

⚠ <https://github.com/free5gc/gtp5g>

Le package **linux-headers-`uname -r`** doit exister !

Prérequis (Déc. 2023) / mode Opérateur

suivre <https://github.com/free5gc/gtp5g>

Suivre les commandes suivantes:

```
git clone https://github.com/free5gc/gtp5g.git && cd gtp5g
make clean && make
```

Commandes Modules :

```
# modinfo udp_tunnel
filename:      /lib/modules/5.15.0-76-generic/kernel/net/ipv4/udp_tunnel.ko
license:      GPL
license:      GPL
srcversion:    4BB7341A13767E0DBF73A20
depends:
retpoline:    Y
intree:       Y
name:         udp_tunnel
vermagic:     5.15.0-76-generic SMP mod_unload modversions
sig_id:       PKCS#7
signer:       Build time autogenerated kernel key
sig_key:      34:18:5F:0B:C9:D8:70:BD:91:78:CF:85:0F:BC:DA:EF:0F:4E:60:FB
sig_hashalgo: sha512
signature:    24:E8:7F:95:9C:BB:FF:8A:F7:BD:C4:4C:2C:58:78:7B:48:94:9E:79:
              A1:23:13:6C:FA:FB:2D:D6:6D:E7:2C:A1:7C:24:03:90:DC:A5:F9:05:
              64:4D:33:0F:18:E0:FA:A7:7A:BE:67:9F:F3:AD:BE:6B:3D:3D:B2:56:
              DF:C4:7C:AF:EE:3D:36:DF:37:B8:31:F9:F8:07:04:07:B4:04:BD:66:
              ...
              8A:80:EA:88:C2:EA:33:45:F7:D2:3C:36:D5:EA:75:23:03:5A:62:B6:
              44:0C:39:DA:05:92:64:46:7A:9E:2C:20:46:51:C4:D8:B2:C5:CC:09:
              3E:2D:5B:A6:6A:02:D1:B4:51:AF:D7:07:DC:B3:0B:AA:0A:BB:B2:3F:
              DE:8B:62:92:DD:CE:60:83:CB:7D:CD:4B
```

Prérequis (Déc. 2023) / mode Opérateur

suivre <https://github.com/free5gc/gtp5g>

Commandes Modules :

```
# modinfo gtp5g
filename:      /lib/modules/5.15.0-76-generic/kernel/drivers/net/gtp5g.ko
alias:         net-pf-16-proto-16-family-gtp5g
alias:         rtnl-link-gtp5g
version:       0.8.3
description:   Interface for 5G GTP encapsulated traffic
author:        Muthuraman <muthuramane.cs03g@g2.nctu.edu.tw>
author:        Yao-Wen Chang <yaowenowo@gmail.com>
license:       GPL
srcversion:    5A86308DFFB318255D09DF6
depends:        udp_tunnel
retpoline:     Y
name:          gtp5g
vermagic:      5.15.0-76-generic SMP mod_unload modversions
```

```
# lsmod | grep gtp5g
gtp5g          118784  0
udp_tunnel    20480  2 gtp5g,sctp
# lsmod | grep udp_tunnel
ip6_udp_tunnel 16384  1 sctp
udp_tunnel     20480  2 gtp5g,sctp
```

```
# tail /lib/modules/5.15.0-76-generic/modules.dep | grep gtp5g
kernel/drivers/net/gtp5g.ko: kernel/net/ipv4/udp_tunnel.ko
```

Installation (Déc. 2023) / mode Opérateur

suivre [GitHub - free5gc/free5gc-compose](https://github.com/free5gc/free5gc-compose)

```
apt update
apt upgrade
git clone https://github.com/free5gc/free5gc-compose.git
cd free5gc-compose/
cd base/
git clone https://github.com/free5gc/free5gc-compose.git
cd ..
make all
docker images
docker compose -f docker-compose-build.yaml build
```

```
docker images
docker pull free5gc/ueransim
docker pull mongo
docker images
```

```
docker pull free5gc/amf
docker pull free5gc/ausf
docker pull free5gc/nrf
docker pull free5gc/pcf
docker pull free5gc/pcf
docker pull free5gc/smf
docker pull free5gc/udm
docker pull free5gc/udr
docker pull free5gc/upf
docker pull free5gc/webui
```

```
docker images
docker-compose -f docker-compose.yaml up -d
docker ps
```

Docker principe / Notion de conteneurs

Offre des moyens d'isoler les ressources et processus pour la virtualisation

- Réserve de ressources
- notion de « namespaces »

Permet de restreindre les accès et la visibilité de ces ressources.

Les ressources sont :

- CPU (puissance de traitements)
- Réseau (connectivités / interfaces et isolation)
- Mémoire (espace de stockage et isolation)

Les ressources sont partageables entre applications distinctes

Les conteneurs offrent des moyens de virtualisations (machines virtuelles ou VMs) et d'isolation des applications très avantageux

- Ils sont légers (se comptent en Mo au VM (quelques Go))
- Possède le strict nécessaire pour les besoins des applications (pas de dépendances) / environnements multiples
- Simplicité et rapidité pour la remise en route (déploiement)
- Duplication / passage à l'échelle

Commandes Docker / Extrait "--help"

```
# docker --help
```

```
Usage:  docker [OPTIONS] COMMAND
```

```
A self-sufficient runtime for containers
```

Common Commands:

run	Create and run a new container from an image
exec	Execute a command in a running container
ps	List containers
build	Build an image from a Dockerfile
pull	Download an image from a registry
push	Upload an image to a registry
images	List images
login	Log in to a registry
logout	Log out from a registry
search	Search Docker Hub for images
version	Show the Docker version information
info	Display system-wide information

Management Commands:

builder	Manage builds
buildx*	Docker Buildx (Docker Inc., v0.11.2)
compose*	Docker Compose (Docker Inc., v2.21.0)
container	Manage containers
context	Manage contexts
image	Manage images
manifest	Manage Docker image manifests and manifest lists
network	Manage networks
plugin	Manage plugins
scan*	Docker Scan (Docker Inc., v0.23.0)
system	Manage Docker
trust	Manage trust on Docker images
volume	Manage volumes

Swarm Commands:

swarm	Manage Swarm
-------	--------------

Commandes Docker / Extrait "--help"

```
Commands:
attach      Attach local standard input, output, and error streams to a running container
commit      Create a new image from a container's changes
cp          Copy files/folders between a container and the local filesystem
create      Create a new container
diff        Inspect changes to files or directories on a container's filesystem
events      Get real time events from the server
export      Export a container's filesystem as a tar archive
history     Show the history of an image
import      Import the contents from a tarball to create a filesystem image
inspect     Return low-level information on Docker objects
kill        Kill one or more running containers
load        Load an image from a tar archive or STDIN
logs        Fetch the logs of a container
pause       Pause all processes within one or more containers
port        List port mappings or a specific mapping for the container
rename      Rename a container
restart     Restart one or more containers
rm          Remove one or more containers
rmi         Remove one or more images
save        Save one or more images to a tar archive (streamed to STDOUT by default)
start       Start one or more stopped containers
stats       Display a live stream of container(s) resource usage statistics
stop        Stop one or more running containers
tag         Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
top         Display the running processes of a container
unpause     Unpause all processes within one or more containers
update      Update configuration of one or more containers
wait        Block until one or more containers stop, then print their exit codes

Global Options:
--config string      Location of client config files (default "/root/.docker")
-c, --context string Name of the context to use to connect to the daemon (overrides DOCKER_HOST env var and default context set with "docker")
-D, --debug          Enable debug mode
-H, --host list       Daemon socket to connect to
-l, --log-level string Set the logging level ("debug", "info", "warn", "error", "fatal") (default "info")
--tls               Use TLS; implied by --tlsverify
--tlscacert string   Trust certs signed only by this CA (default "/root/.docker/ca.pem")
--tlscert string      Path to TLS certificate file (default "/root/.docker/cert.pem")
--tlskey string       Path to TLS key file (default "/root/.docker/key.pem")
--tlsverify          Use TLS and verify the remote
-v, --version         Print version information and quit

Run 'docker COMMAND --help' for more information on a command.

For more help on how to use Docker, head to https://docs.docker.com/go/guides/
```

Docker compose

Découverte

```
docker ps
version: '3.8'
```

```
services:
  free5gc-upf:
    container_name: upf
    image: free5gc/upf:v3.3.0
    command: bash -c ". /upf-iptables.sh && ./upf -c ./config/upfcfg.yaml"
    volumes:
      - ./config/upfcfg.yaml:/free5gc/config/upfcfg.yaml
      - ./config/upf-iptables.sh:/free5gc/upf-iptables.sh
    cap_add:
      - NET_ADMIN
    networks:
      privnet:
        aliases:
          - upf.free5gc.org
```

```
db:
  container_name: mongodb
  image: mongo
  command: mongod --port 27017
  expose:
    - "27017"
  volumes:
    - dbdata:/data/db
  networks:
    privnet:
      aliases:
        - db
```

```
free5gc-nrf:
  container_name: nrf
  image: free5gc/nrf:v3.3.0
  command: ./nrf -c ./config/nrfcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/nrfcfg.yaml:/free5gc/config/nrfcfg.yaml
  environment:
    DB_URI: mongodb://db/free5gc
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - nrf.free5gc.org
  depends_on:
    - db
```

```
free5gc-amf:
  container_name: amf
  image: free5gc/amf:v3.3.0
  command: ./amf -c ./config/amfcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/amfcfg.yaml:/free5gc/config/amfcfg.yaml
  environment:
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - amf.free5gc.org
  depends_on:
    - free5gc-nrf
```

```
free5gc-smf:
  container_name: smf
  image: free5gc/smf:v3.3.0
  command: ./smf -c ./config/smfcfg.yaml -u ./config/uerouting.yaml
  expose:
    - "8000"
  volumes:
    - ./config/smfcfg.yaml:/free5gc/config/smfcfg.yaml
    - ./config/uerouting.yaml:/free5gc/config/uerouting.yaml
  environment:
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - smf.free5gc.org
  depends_on:
    - free5gc-nrf
    - free5gc-upf
```

```
free5gc-pcf:
  container_name: pcf
  image: free5gc/pcf:v3.3.0
  command: ./pcf -c ./config/pcfcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/pcfcfg.yaml:/free5gc/config/pcfcfg.yaml
  environment:
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - pcf.free5gc.org
  depends_on:
    - free5gc-nrf
```

Docker compose

Découverte

```
free5gc-nssf:
  container_name: nssf
  image: free5gc/nssf:v3.3.0
  command: ./nssf -c ./config/nssfcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/nssfcfg.yaml:/free5gc/config/nssfcfg.yaml
  environment:
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - nssf.free5gc.org
  depends_on:
    - free5gc-nrf

free5gc-n3iwf:
  container_name: n3iwf
  image: free5gc/n3iwf:v3.3.0
  command: sh -c ". /n3iwf-ipsec.sh && ./n3iwf -c ./config/n3iwfcfg.yaml"
  volumes:
    - ./config/n3iwfcfg.yaml:/free5gc/config/n3iwfcfg.yaml
    - ./config/n3iwf-ipsec.sh:/free5gc/n3iwf-ipsec.sh
  environment:
    GIN_MODE: release
  cap_add:
    - NET_ADMIN
  networks:
    privnet:
      aliases:
        - n3iwf.free5gc.org
  depends_on:
    - free5gc-amf
    - free5gc-smf
    - free5gc-upf
```

```
free5gc-ausf:
  container_name: ausf
  image: free5gc/ausf:v3.3.0
  command: ./ausf -c ./config/ausfcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/ausfcfg.yaml:/free5gc/config/ausfcfg.yaml
  environment:
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - ausf.free5gc.org
  depends_on:
    - free5gc-nrf

free5gc-udm:
  container_name: udm
  image: free5gc/udm:v3.3.0
  command: ./udm -c ./config/udmcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/udmcfg.yaml:/free5gc/config/udmcfg.yaml
  environment:
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - udm.free5gc.org
  depends_on:
    - db
    - free5gc-nrf

free5gc-udr:
  container_name: udr
  image: free5gc/udr:v3.3.0
  command: ./udr -c ./config/udrcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/udrcfg.yaml:/free5gc/config/udrcfg.yaml
  environment:
    DB_URI: mongodb://db/free5gc
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - udr.free5gc.org
  depends_on:
    - db
    - free5gc-nrf
```

Docker compose

Découverte

```
free5gc-webui:
  container_name: webui
  image: free5gc/webui:v3.3.0
  command: ./webui -c ./config/webuicfg.yaml
  expose:
    - "2122"
    - "2121"
  volumes:
    - ./config/webuicfg.yaml:/free5gc/config/webuicfg.yaml
  environment:
    - GIN_MODE=release
  networks:
    privnet:
      aliases:
        - webui
  ports:
    - "5000:5000"
    - "2122:2122"
    - "2121:2121"
  depends_on:
    - db

ueransim:
  container_name: ueransim
  image: free5gc/ueransim:latest
  command: ./nr-gnb -c ./config/gnbcfg.yaml
  volumes:
    - ./config/gnbcfg.yaml:/ueransim/config/gnbcfg.yaml
    - ./config/uecfg.yaml:/ueransim/config/uecfg.yaml
  cap_add:
    - NET_ADMIN
  devices:
    - "/dev/net/tun"
  networks:
    privnet:
      aliases:
        - gnb.free5gc.org
  depends_on:
    - free5gc-amf
    - free5gc-upf
```

```
free5gc-chf:
  container_name: chf
  image: free5gc/chf@sha256:8b29aeab340[...]38903c8f4d7e1e9677
  command: ./chf -c ./config/chfcfg.yaml
  expose:
    - "8000"
  volumes:
    - ./config/chfcfg.yaml:/free5gc/config/chfcfg.yaml
  environment:
    DB_URI: mongodb://db/free5gc
    GIN_MODE: release
  networks:
    privnet:
      aliases:
        - chf.free5gc.org
  depends_on:
    - db
    - free5gc-nrf
    - free5gc-webui

networks:
  privnet:
    ipam:
      driver: default
      config:
        - subnet: 10.100.200.0/24
    driver_opts:
      com.docker.network.bridge.name: br-free5gc

volumes:
  dbdata:
```

Ueransim configuration

Découverte

gnbcfg.yaml

```
mcc: '208'          # Mobile Country Code value
mnc: '93'           # Mobile Network Code value (2 or 3 digits)

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

linkIp: 127.0.0.1   # gNB's local IP address for Radio Link Simulation (Usually same with local IP)
ngapIp: gnb.free5gc.org # gNB's local IP address for N2 Interface (Usually same with local IP)
gtpIp: gnb.free5gc.org   # gNB's local IP address for N3 Interface (Usually same with local IP)

# List of AMF address information
amfConfigs:
  - address: amf.free5gc.org
    port: 38412

# List of supported S-NSSAIs by this gNB
slices:
  - sst: 0x1
    sd: 0x010203

# Indicates whether or not SCTP stream number errors should be ignored.
ignoreStreamIds: true
(END)
```

Ueransim configuration

uecfg.yaml

```
# IMSI number of the UE. IMSI = [MCC|MNC|MSISDN] (In total 15 digits)
supi: 'imsi-208930000000001'
# Mobile Country Code value of HPLMN
mcc: '208'
# Mobile Network Code value of HPLMN (2 or 3 digits)
mnc: '93'

# Permanent subscription key
key: '8baf473f2f8fd09487cccbd7097c6862'
# Operator code (OP or OPC) of the UE
op: '8e27b6af0e692e750f32667a3b14605d'
# This value specifies the OP type and it can be either 'OP' or 'OPC'
opType: 'OPC'
# Authentication Management Field (AMF) value
amf: '8000'
# IMEI number of the device. It is used if no SUPI is provided
imei: '356938035643803'
# IMEISV number of the device. It is used if no SUPI and IMEI is provided
imeiSv: '4370816125816151'

# List of gNB IP addresses for Radio Link Simulation
gnbSearchList:
  - 127.0.0.1

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

# UAC Access Control Class
uacAcc:
  normalClass: 0
  class11: false
  class12: false
```

```
class13: false
class14: false
class15: false

# Initial PDU sessions to be established
sessions:
  - type: 'IPv4'
    apn: 'internet'
    slice:
      sst: 0x01
      sd: 0x010203

# Configured NSSAI for this UE by HPLMN
configured-nssai:
  - sst: 0x01
    sd: 0x010203

# Default Configured NSSAI for this UE
default-nssai:
  - sst: 1
    sd: 1


# Supported integrity algorithms by this UE
integrity:
  IA1: true
  IA2: true
  IA3: true


# Supported encryption algorithms by this UE
ciphering:
  EA1: true
  EA2: true
  EA3: true


# Integrity protection maximum data rate for user
plane
integrityMaxRate:
  uplink: 'full'
  downlink: 'full'
```


MongoDB (port 5000)


- <http://127.0.0.1:5000/#/ueinfo/imsi-2089300000000003>

Log out

 REALTIME STATUS

 SUBSCRIBERS


 ANALYTICS


 TENANT AND USER


Subscribers


New Subscriber


PLMN	UE ID		
20893	imsi-2089300000000001	<button>Delete</button>	<button>Modify</button>
20893	imsi-2089300000000002	<button>Delete</button>	<button>Modify</button>
20893	imsi-2089300000000003	<button>Delete</button>	<button>Modify</button>

Log out

 REALTIME STATUS

 SUBSCRIBERS

 ANALYTICS

 TENANT AND USER

Registered UEs

Refresh

SUPI ▼▲	Status ▼▲	Details
imsi-2089300000000003	CONNECTED	<button>Show Info</button>

MongoDB (port 5000)

- <http://127.0.0.1:5000/#/ueinfo/imsi-208930000000003>

AMF Information [SUPI:imsi-208930000000003]

Information Entity	Value
AccessType	3GPP_ACCESS
CmState	CONNECTED
Guti	20893cafe00000000001
Mcc	208
Mnc	93
Supi	imsi-208930000000003
Tac	000001
Dnn	internet
PduSessionId	1
Sd	010203
SmContextRef	urn:uuid:934a0589-1c2d-4641-94cd-f54db9eff802
Sst	1

MongoDB (port 5000)

- <http://127.0.0.1:5000/#/ueinfo/imsi-208930000000003>

SMF Information [SUPI:imsi-208930000000003]

Information Entity	Value
AnType	3GPP_ACCESS
Dnn	internet
LocalSEID	
PDUAddress	10.60.0.1
PDUSessionID	1
RemoteSEID	
Sd	010203
Sst	1

MongoDB (port 5000)

- <http://127.0.0.1:5000/#/ueinfo/imsi-208930000000003>

AMF Information [SUPI:imsi-2089300007487]

Information Entity	Value
AccessType	3GPP
Supi	imsi-2089300007487
Guti	guti-2089300007487
mcc	123
mnc	456
tac	1
CmState	string
PduSessionId	int
smContextRef	string
sst	int
sd	string
Dnn	internet

MongoDB (port 5000)

- <http://127.0.0.1:5000/#/ueinfo/imsi-208930000000003>

SMF Information [SUPI:imsi-2089300007487]

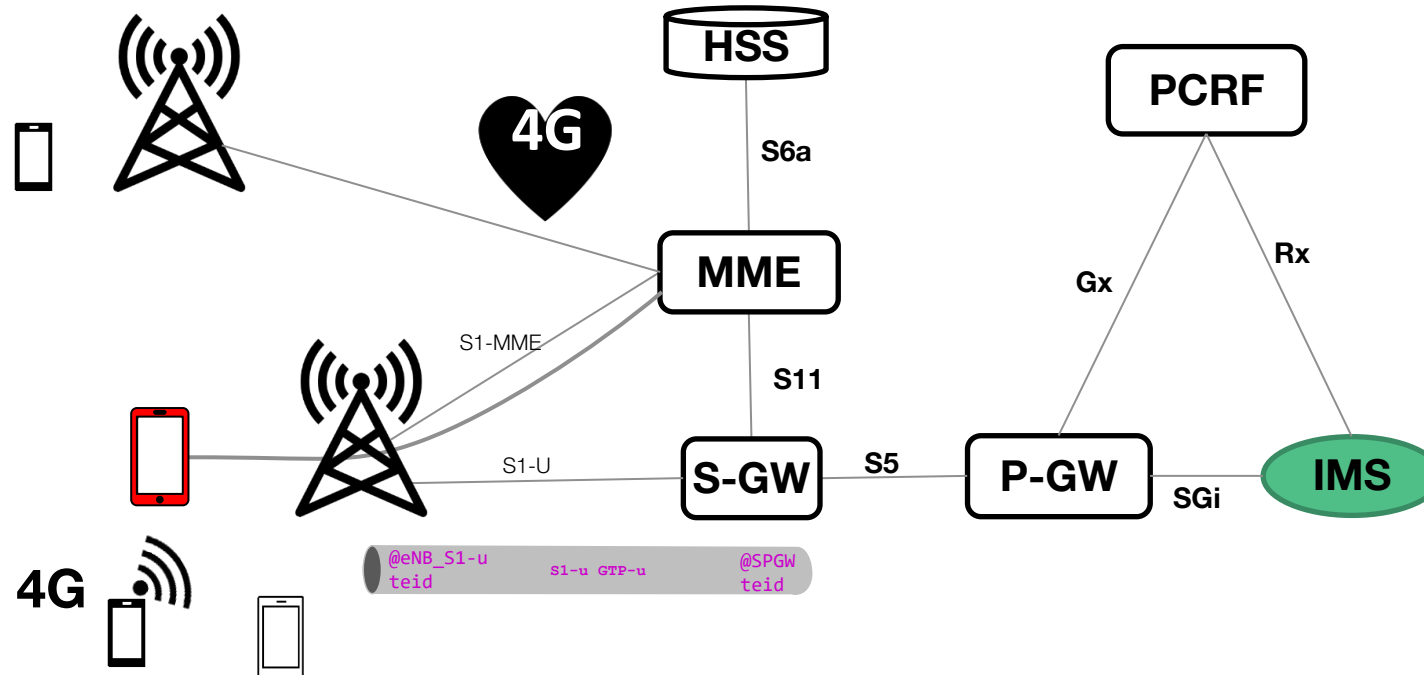
Information Entity	Value
AnType	
Dnn	
LocalSEID	
PDUAddress	
PDUSessionID	
RemoteSEID	
Sd	
Sst	

EPC & 5G

Architecture et Call Flows (Analyse traces wireshark)

Suzanne Chan (Déc. 2023/ Janv. 2024)

E-UTRAN/EPC (4G) Network

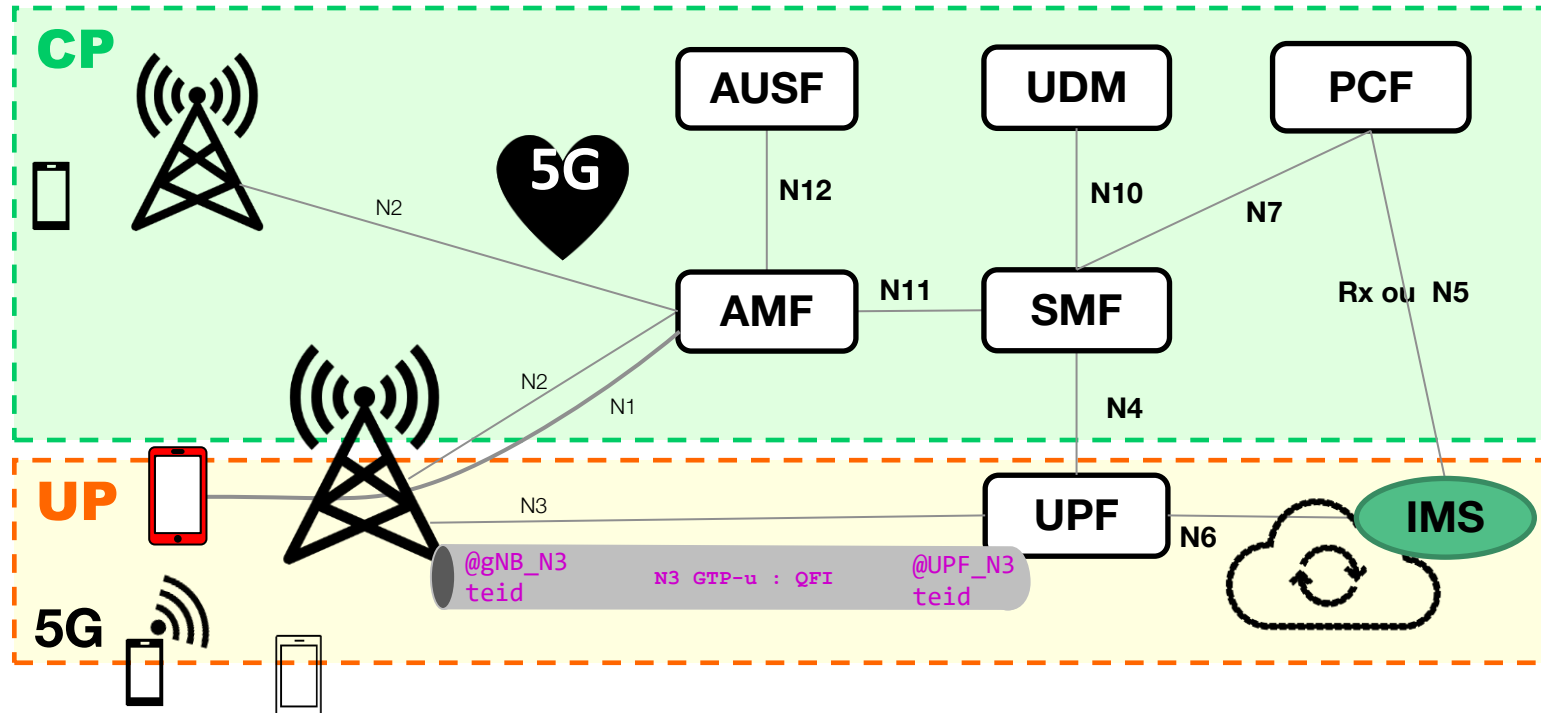


3rd Generation Partnership Project
Technical Specification Group Core Network and Terminal
<https://www.3gpp.org/3gpp-groups/core-network-terminals-ct/ct-wg1>
EPC : Evolved Packet Core Network

4G

HSS : Home Subscriber Server
MME : Mobility Management Entity
S-GW : Serving Gateway
P-GW : Packet Gateway
PCRF : Policy & Charging Rules Function
IMS : IP Multimedia Subsystem

5G Network



AUSF : Authentication Server Function
UDR : Unified Data Repository
UDM : Unified Data Management

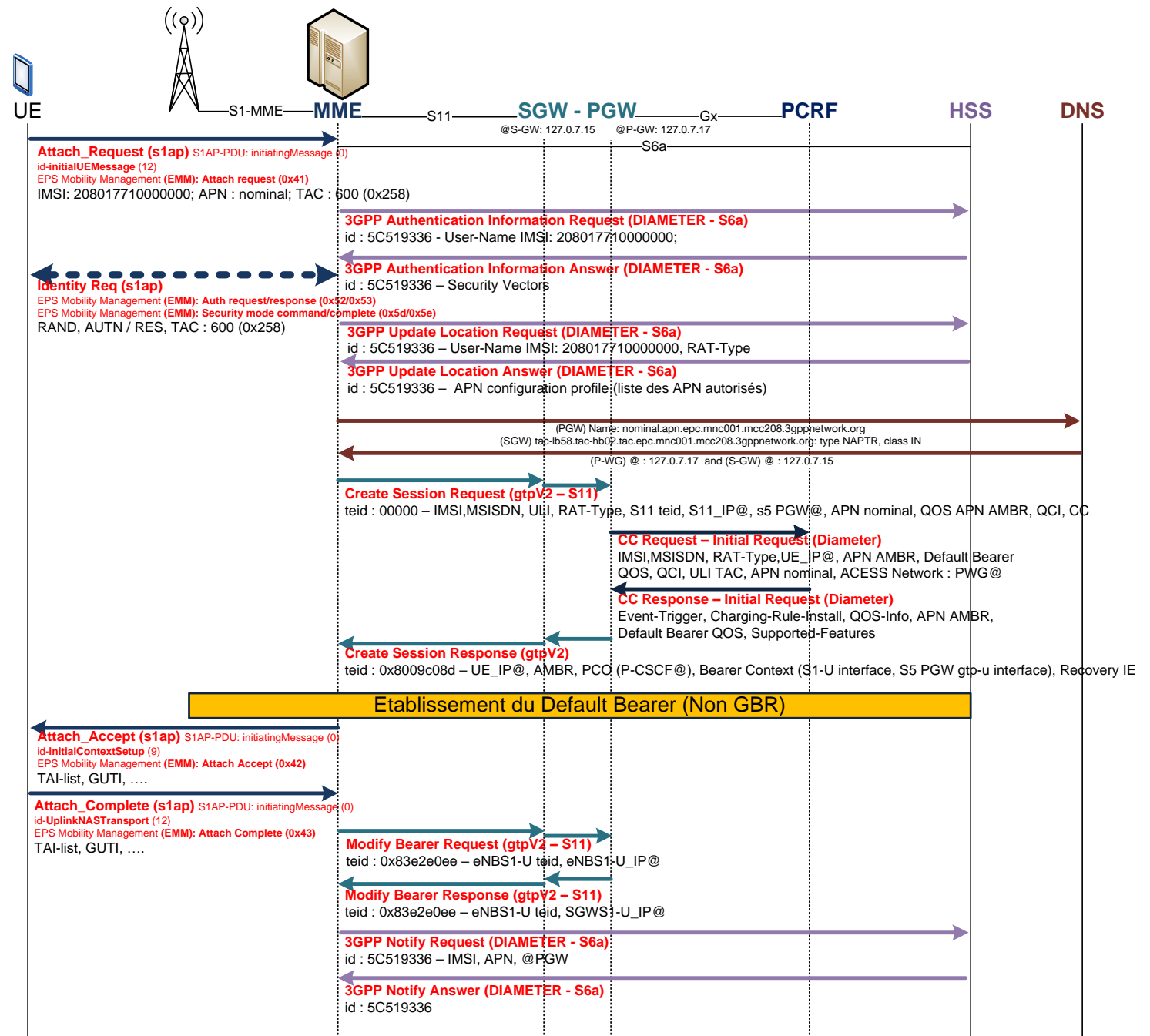
CP : Control Plane
UP : User Plane

5G

AMF : Access and Mobility Management Function
SMF : Session Management Function
UPF : User Plane Function
PCF : Policy & Charging Function
IMS : IP Multimedia Subsystem (service Voix sur IP)

Call flow 4G

Default Bearer Internet ...



Call flow 5G
à construire à l'aide
des captures wireshark

NOTE pour le rapport de synthèse

M2 RSA (Déc. 2023/ Janv. 2024)

Éléments du rapport de synthèse

Cette présentation constitue une aide précieuse pour la réalisation de ce rapport de synthèse.

Vous êtes un nouvel acteur et vous voulez entrer dans ce nouveau marché des télécommunications... Vous étudiez la faisabilité avec la solution open source « free5gc » mode docker car léger, rapide et simple.

*Avec la rapidité de déploiement de l'infrastructure free5gc, vous êtes invité à redéploier cette infrastructure sur votre propre réseau privé ... Vous devez l'adapter à votre propre environnement donc sous réseau (par exemple le 127.5.10.0/23). Vous devez également changer le code MNC opérateur qui n'est plus le 93 mais **50** par exemple.*

Pour le redéploiement et les captures d'écran, vous pouvez refaire les exercices avec ce nouveau réseau ou un autre de votre choix (un différent de celui fourni par défaut 10.100.200.0/24).

Vous pouvez répartir par étapes, détaillées ci-dessous en exemple:

Introduction: le nom de la société, l'objet de l'étude ...

Installation: Problème d'installation et contournement pour résoudre ...

Déploiement: étude de l'infrastructure, nombre de gNBs, nombre d'équipements mobiles, cas de panne ... et identification de cas de panne et résolutions ...

Conception: Service applicatif sur UDP/IP application de transport de la voix / (ou du menu du jour du restaurant U à l'UPC), étude de faisabilité concrétisée par le schéma de routage demandé.