共进自研 5G 核心网环境配置

文件编号：	文件页数：第　页 共　　页	版本号：
文件名称：5G 核心网环境配置 使用限制：需要运行 ubuntu-18.04 系统 下载地址：http://mirrors.aliyun.com/ubuntu-releases/18.04/		

更改记录 Release note

版本号	修改描述	更改部门	更改人	审核人	批准人	生效日期
V1.0	首板发行	软件二部	刘俭	田野	田野	2021.3.18
V1.1	修改	测试部	何文亮			2023.4.13
V1.2	增加转发命令	测试部	何文亮			2023.9.18
V1.3	增加 openssh 方法和 sim 卡固定 ip	测试部	何文亮			2023.10.7
V1.4	增加 4G 核心网的配置和启动	测试部	郭锁奇			2023.10.11
V1.5	增加新版 Ubuntu 核心网启动报错处理方法	测试部	李智涌			2024.10.16

目录

一、 环境配置说明

1. 环境安装

1.安装 mongodb

$ sudo apt update

$ sudo apt install mongodb

$ sudo systemctl start mongodb (if '/usr/bin/mongod' is not running)

$ sudo systemctl enable mongodb (ensure to automatically start it on system boot)

2.安装核心网依赖的软件

```
$ sudo apt install python3-pip python3-setuptools python3-wheel ninja-build build-essential flex bison git libsctp-dev libgnutls28-dev libgcrypt-dev libssl-dev libidn11-dev libmongoc-dev libbson-dev libyaml-dev libnghttp2-dev libmicrohttpd-dev libcurl4-gnutls-dev libnghttp2-dev libtins-dev meson
```

3.安装工具软件(缺啥安装啥)

$ sudo apt install vim tar

4. 将 5gc.tar.gz 拷贝到/usr/local 目录下并解压。

sudo tar -xzvf 5gc.tar.gz

5.安装 wireshark

因为 ubuntu 18.04 自带的 wireshark 版本太低，不支持 ngap 协议，所以需要升级为高版本

① https://blog.csdn.net/valecalida/article/details/103579778 安装 wireshark 的操作方法

② 如果安装过程库文件失败，请修改 ubuntu18.04 的更新源以后，重新进行安装。(源配置可以混用)

更新源方法:https://www.cnblogs.com/2sheep2simple/p/10610847.html

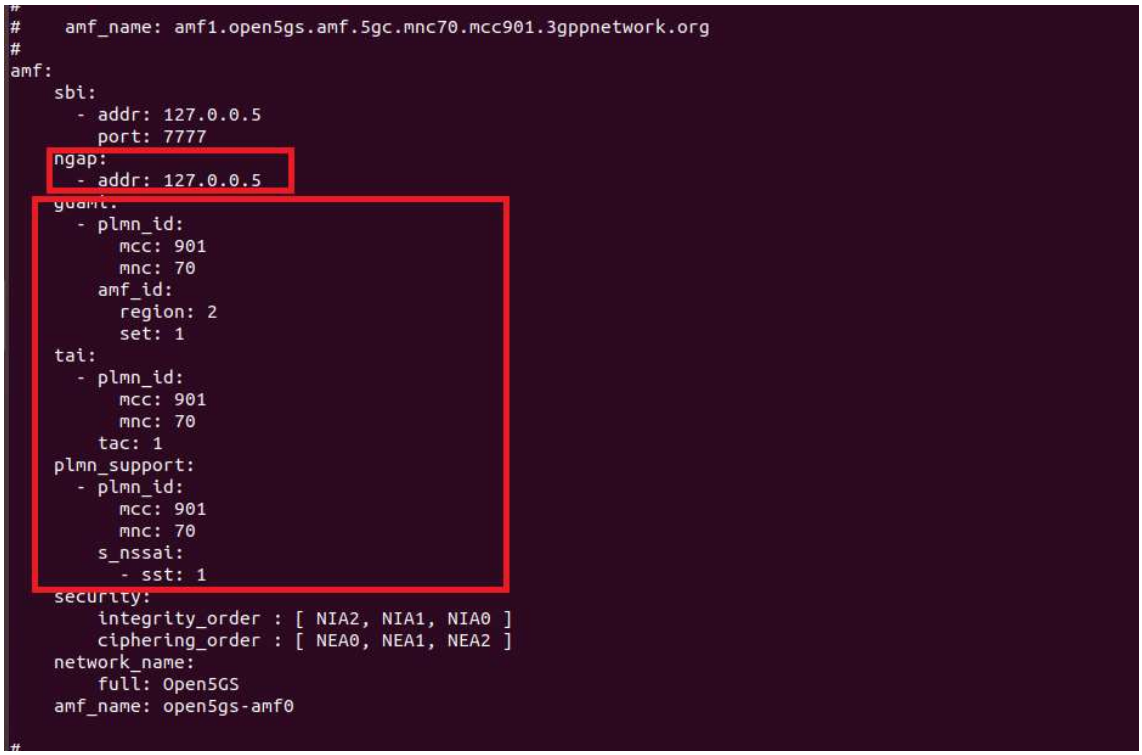
说明：以上安装需要将设备联网，可直接命令获取互联网库安装。

2. 5G 核心网的配置和启动

1. 配置

配置文件在 install/etc/5gc/目录下

(1) 修改 amf.yaml

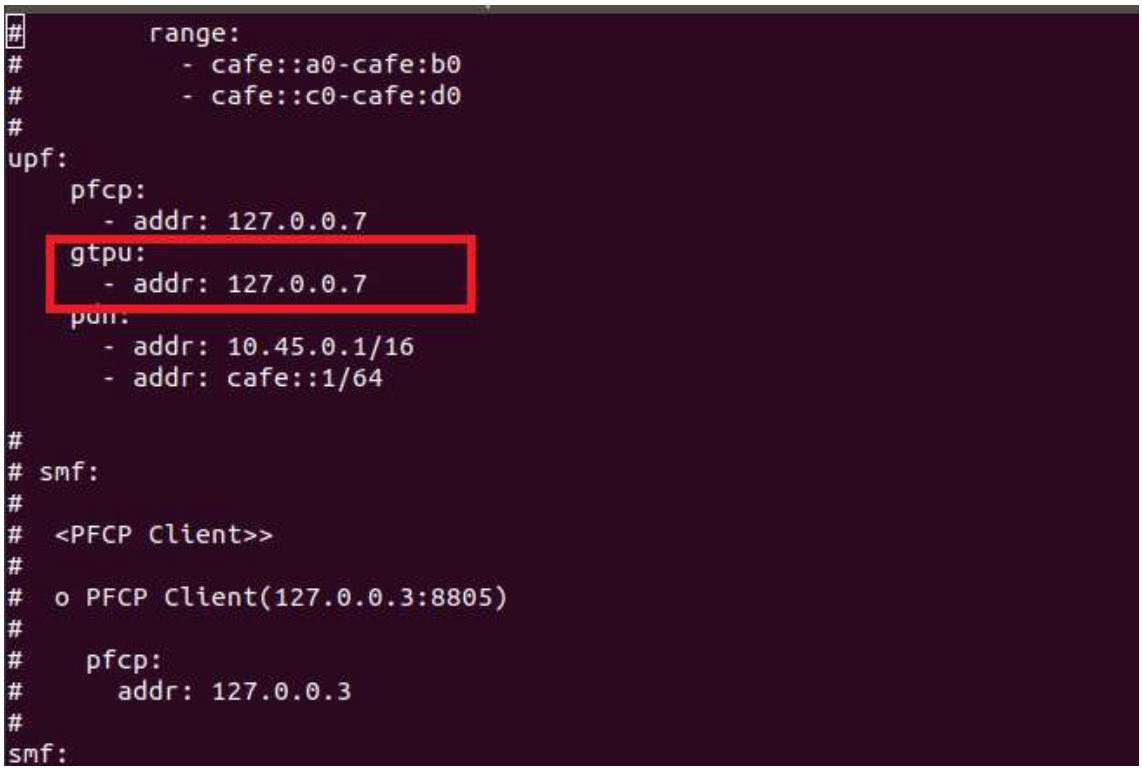


请将以上配置文件的 ngap 的 IP 地址修改为设备的 IP 地址，比如 192.168.106.201。

根据 gNB 的实际配置，修改核心网相关的配置，请修改上图红框部分的值（guami，tai 和 plmn 节点）。

主要修改 mcc 和 mnc（与 sim 卡一致）, tac(与基站一致)。

(2)修改 upf.yaml

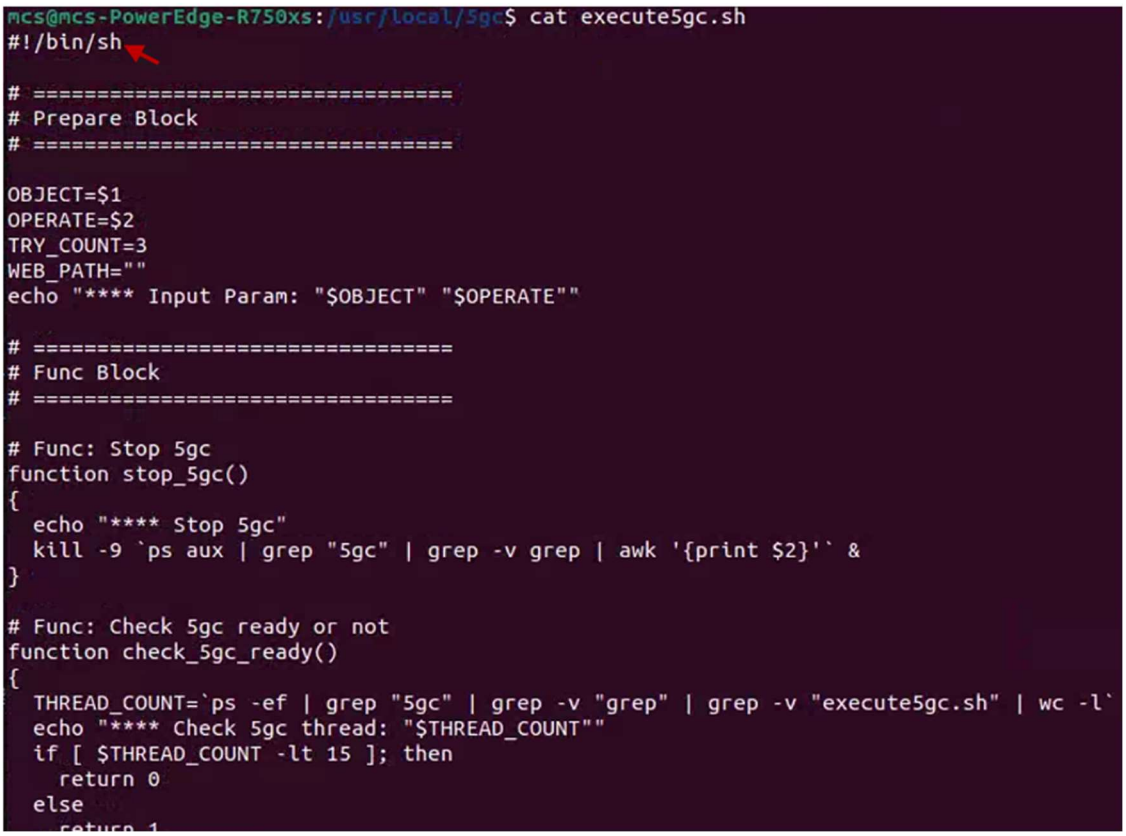


请将以上配置文件的 gtpu 的 IP 地址修改为设备的 IP 地址，比如 192.168.106.201。

2. 核心网的启动和运行

进入到核心网的目录：cd /usr/local/5gc

修改 execute5gc.sh 脚本：修改 sh 为 bash



```
mcs@mcs-PowerEdge-R750xs:/usr/local/5gc$ cat execute5gc.sh
#!/bin/sh

# ================================
# Prepare Block
# ================================

OBJECT=$1
OPERATE=$2
TRY_COUNT=3
WEB_PATH=""
echo "**** Input Param: "$OBJECT" "$OPERATE""

# ================================
# Func Block
# ================================

# Func: Stop 5gc
function stop_5gc()
{
  echo "**** Stop 5gc"
  kill -9 `ps aux | grep "5gc" | grep -v grep | awk '{print $2}'` &
}

# Func: Check 5gc ready or not
function check_5gc_ready()
{
  THREAD_COUNT=`ps -ef | grep "5gc" | grep -v "grep" | grep -v "execute5gc.sh" | wc -l`
  echo "**** Check 5gc thread: "$THREAD_COUNT""
  if [ $THREAD_COUNT -lt 15 ]; then
    return 0
  else
    return 1
```

运行核心网：sudo bash ./execute5gc.sh c r

停止核心网：sudo bash ./execute5gc.sh c s

注意：以上命令的执行在 root 权限下执行。

脚本不能执行的情况：

ls -al /bin/sh

sudo dpkg-reconfigure dash

3. 4G 核心网的配置和启动

1. 配置

配置文件在 install/etc/5gc/目录下

(1) 检查 amf.yaml

请将以上配置文件的 ngap 的 IP 地址还原为默认 IP 地址 127.0.0.5。

```
logger:
    file: /usr/local/5gc/var/log/5gc/amf.log

amf:
    sbi:
      - addr: 127.0.0.5
        port: 7777
    ngap:
      - addr: 127.0.0.5
    guami:
      - plmn_id:
          mcc: 901
          mnc: 70
        amf_id:
          region: 2
          set: 1
    tai:
      - plmn_id:
          mcc: 901
          mnc: 70
        tac: 1
    plmn_support:
      - plmn_id:
```

(2) 检查 upf.yaml

请将以上配置文件的 gtpu 的 IP 地址还原为默认 IP 地址 127.0.0.7。

```
logger:
    file: /usr/local/5gc/var/log/5gc/upf.log

upf:
    pfcp:
      - addr: 127.0.0.7
    gtpu:
      - addr: 127.0.0.7
    subnet:
      - addr: 10.45.0.1/16
      - addr: 2001:230:cafe::1/48

smf:

parameter:

max:

pool:
```

(3) 修改 mme.yaml

请将以上配置文件的 s1ap 的 IP 地址修改为设备的 IP 地址，比如 192.168.50.129。

根据 eNB 的实际配置，修改核心网相关的配置，请修改上图红框部分的值（gummei，tai 和 plmn 节点）。

主要修改 mcc 和 mnc（与 sim 卡一致）, tac(与基站一致)。

```
logger:
    file: /usr/local/5gc/var/log/5gc/mme.log

mme:
    freeDiameter: /usr/local/5gc/etc/freeDiameter/mme.conf
    s1ap:
      - addr: 192.168.50.129
    gtpc:
      - addr: 127.0.0.2
    metrics:
      - addr: 127.0.0.2
        port: 9090
    gummei:
      plmn_id:
        mcc: 460
        mnc: 88
      mme_gid: 2
      mme_code: 1
    tai:
      plmn_id:
        mcc: 460
        mnc: 88
      tac: 1
    security:
        integrity_order : [ EIA2, EIA1, EIA0 ]
        ciphering_order : [ EEA0, EEA1, EEA2 ]
    network_name:
        full: 5GC
    mme_name: 5gc-mme0

sgwc:
    gtpc:
      - addr: 127.0.0.3
```

(4) 修改 sgwu.yaml

请将以上配置文件的 gtpu 的 IP 地址修改为设备的 IP 地址，比如 192.168.50.129。

```
    file: /usr/local/5gc/var/log/5gc/sgwu.log

sgwu:
    pfcp:
      - addr: 127.0.0.6
    gtpu:
      - addr: 192.168.50.129

sgwc:

parameter:

max:

pool:
```

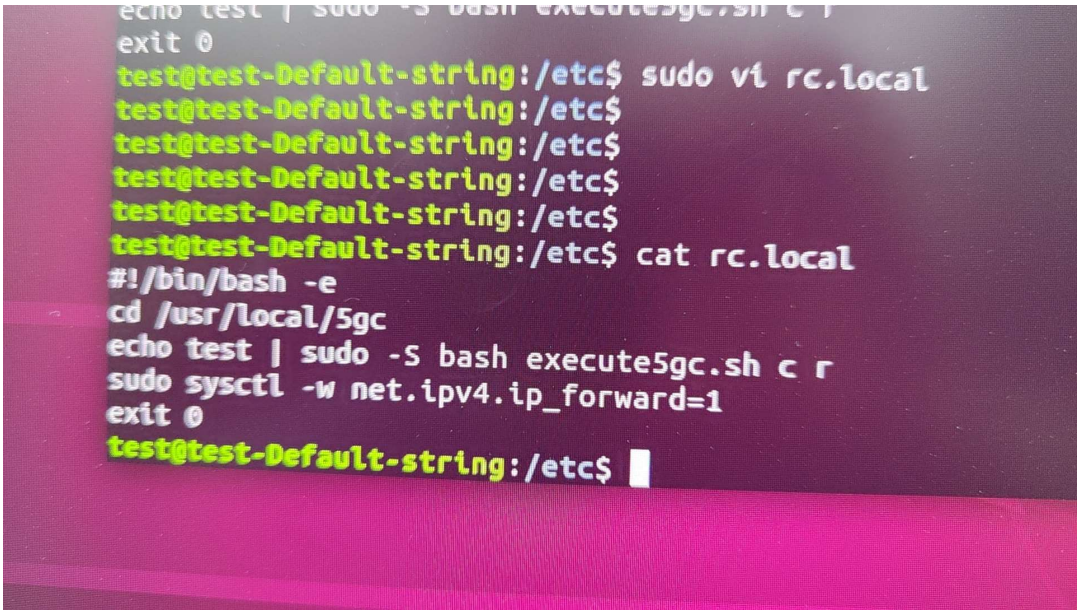
2. 核心网的启动和运行

4G 核心网的启动方法与 5GC 核心网的相同，请参照 5G 核心网启动部分章节。

4. 核心网上业务转发配置

sudo sysctl -w net.ipv4.ip_forward=1

可以将该命令写到 rc.local 文件中开机启动，这样在核心网侧的业务服务器的业务通过核心网转发。



sudo iptables -t nat -A POSTROUTING -s 10.45.0.0/16 ! -o ogstun -j MASQUERADE

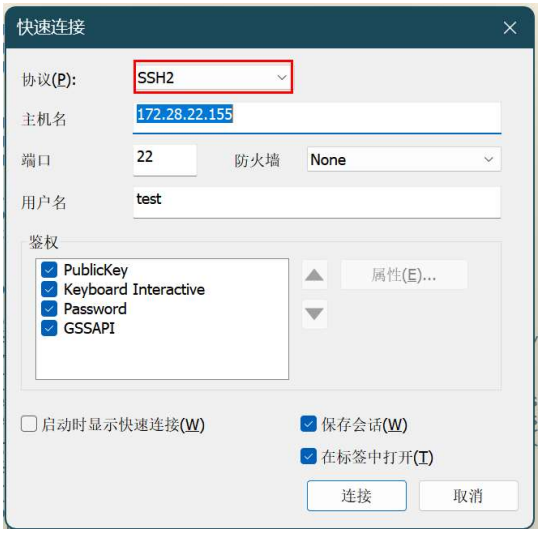
-----如何接入的终端需要连接互联网，核心网需要增加这条转发命令。

5. 安装 openssh 软件方便 ssh 设备进行操作配置

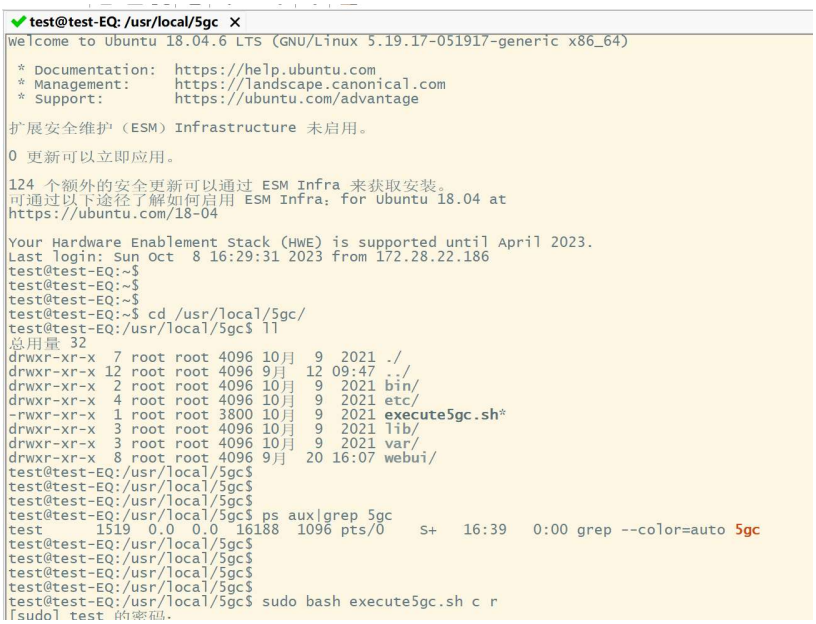
SSH 是一种安全的远程连接方式，可以确保您的连接是加密的，并且只有授权的用户才能访问系统。在 Ubuntu 中，您可以使用 OpenSSH 服务器和客户端来建立 SSH 连接。您需要在 Ubuntu 系统上安装 OpenSSH 服务器。您可以使用以下命令安装：

sudo apt-get install openssh-server

安装后，可使用 Windows 下的支持 ssh 协议的软件登录，比如 SecureCRT，如下：



选择 ssh2 协议，输入核心网设备的 ip 地址，输入用户名，点击连接，弹出输入密码框，输入密码登录，即可 ssh 进入设备，进行命令操作。



二、 开户操作说明

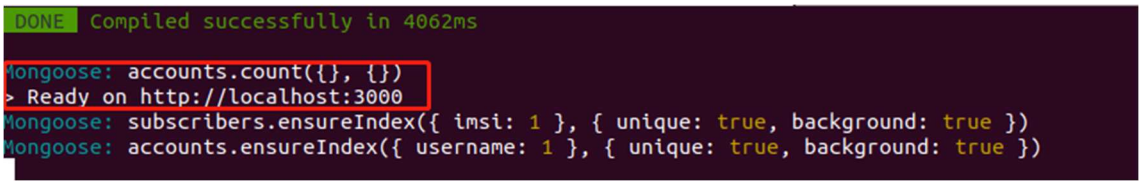
1. WebUI 增加签约用户

(1) 执行如下命令启动 WebUI；

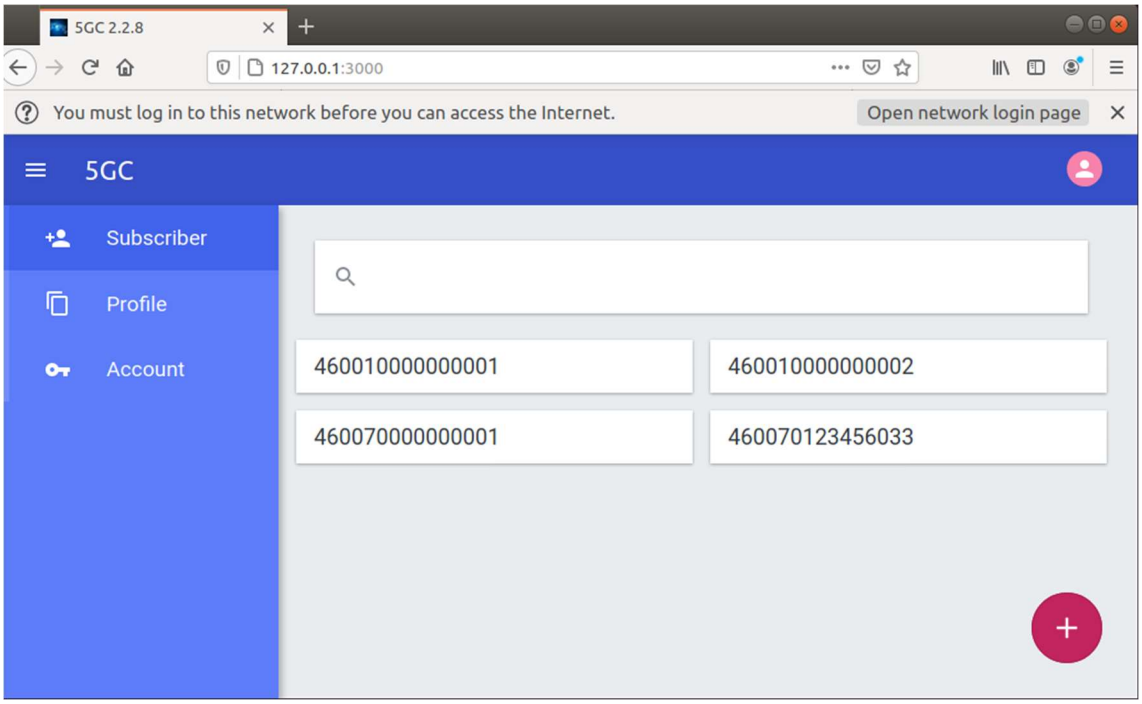
cd /usr/local/5gc/webui

npm run dev

注意：以上命令的执行在 root 权限下执行。



(2) 打开虚拟机自带火狐浏览器，登录地址 http://localhost:3000，用户名密码 admin/1423，如下图所示。



A. 编辑操作：鼠标滑过任意 IMSI，会出现编辑和删除按钮；

B. 增加操作：点击图示右下角加号，填写如下数据即可：

IMSI：自定义，注意 Plmn 值

Subscriber Key(K)*：01020304050607080102030405060708（和烧录白卡一致）；

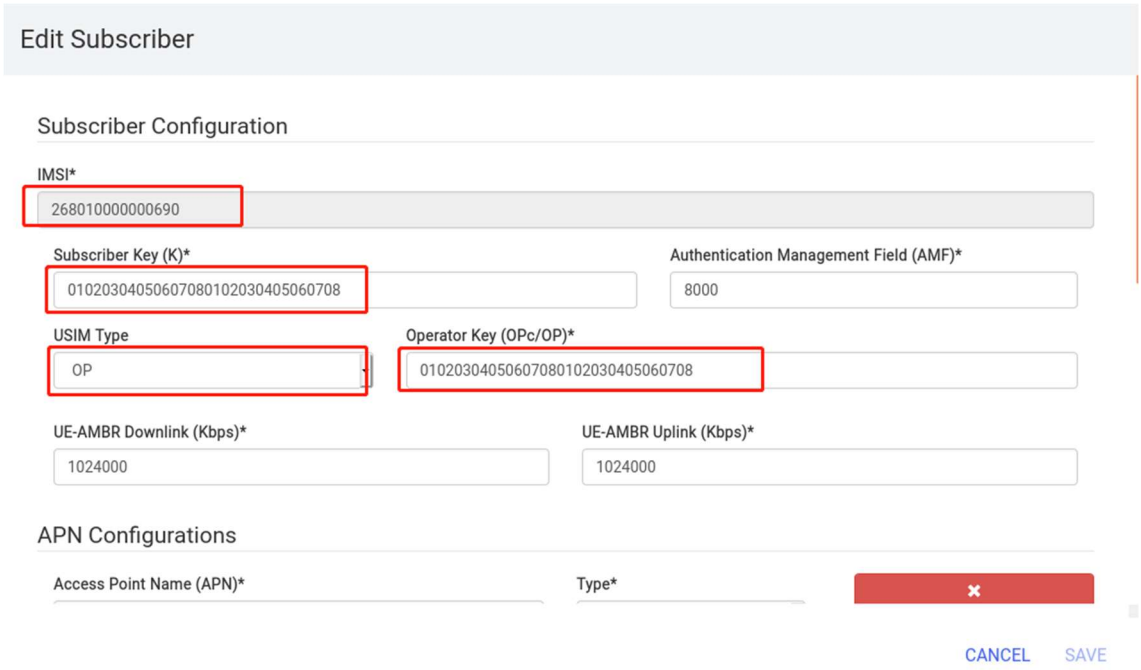
USIM Type：选择 OP（和烧录白卡一致）；

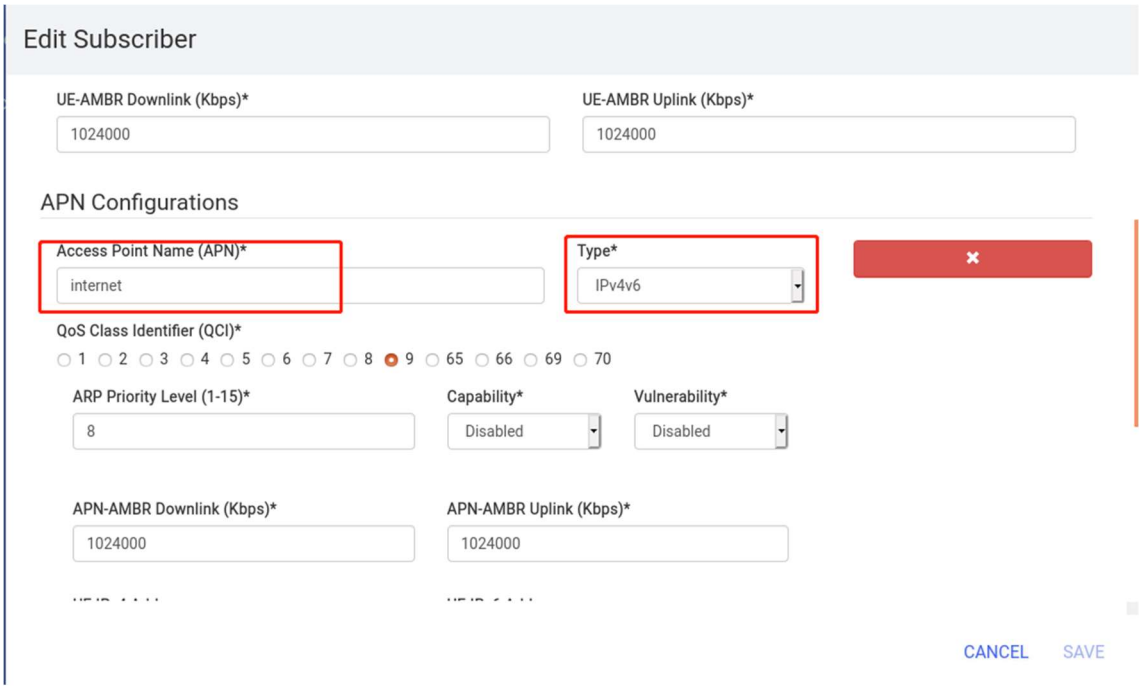
Operator Key (OPc/OP)*：01020304050607080102030405060708（和烧录白卡一致）；

Access Point Name (APN)*：internet（和烧录白卡一致）；

Type：IPv4v6；

C. 点击 Save 保存即可。

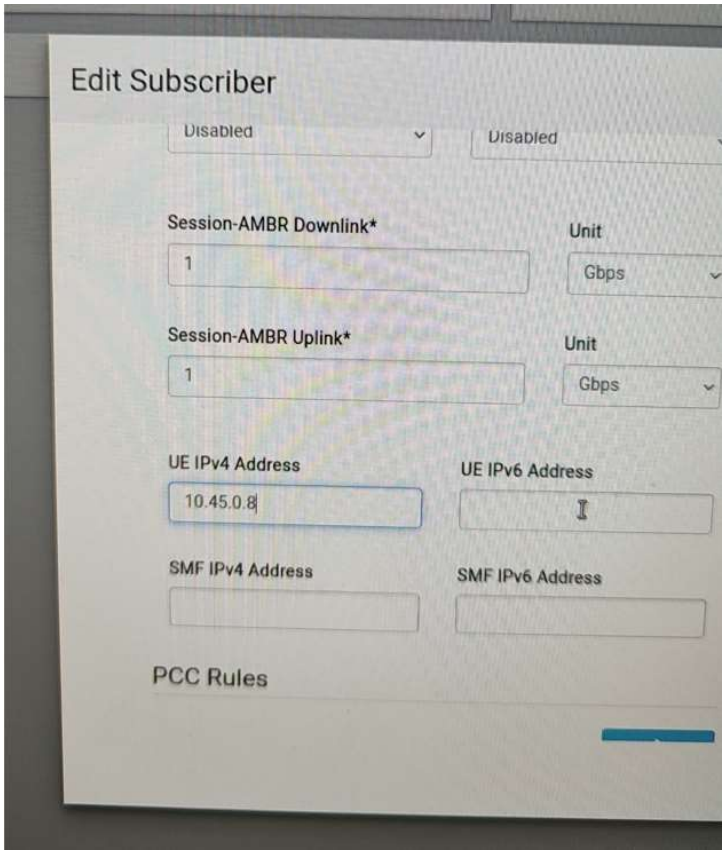




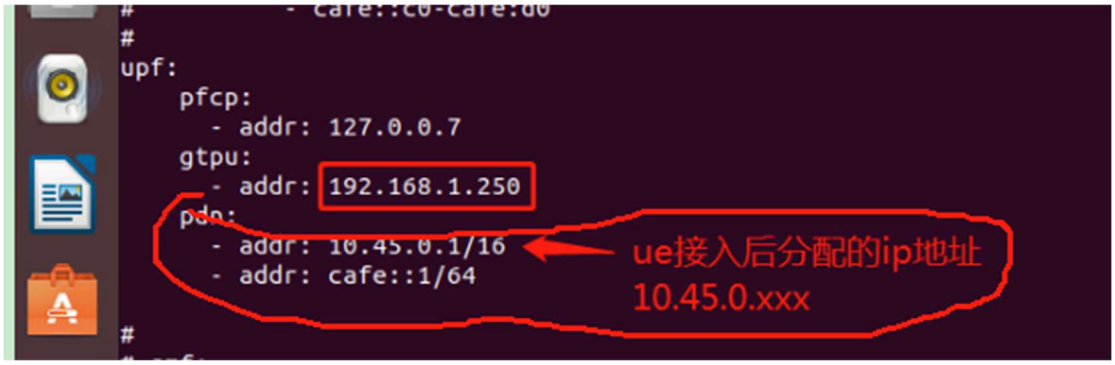
(3) 停止 WebUI，只需要在 terminal 中 Ctrl+C 即可。

2. 给每个 sim 卡分配固定 ip 地址

网页上在 imsi 登记位置，在 ue ipv4 address 位置，输入要为该 sim 卡分配的 ip 地址，保存即可。



注意，分配的 ip 地址不要超过地址池范围。



三、 异常情况

1、安装 Ubuntu18 系统可能会出现花屏，解决办法如下：

安装/开机Ubuntu18.04出现花屏

> 在华硕主板上安装Ubuntu18.04时出现花屏，问题出现好像和显卡有关，我的是RTX 2060S，现将解决方法记录一下

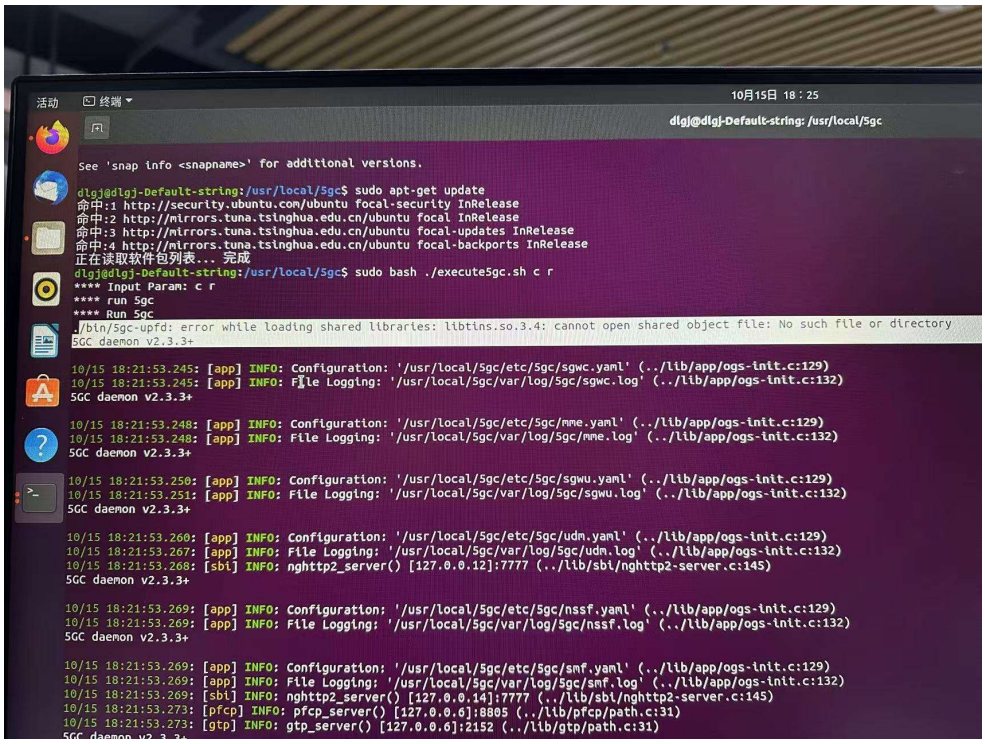
- 安装时花屏：
  在安装GRUB页面的时候选择*install Ubuntu*，不要点击，按**e**进入编辑页面，在**quiet splash**后面删除“---”，添加**nomodeset**以支持nvidia显卡，然后*Ctrl+X*进行安装。
- 开机时花屏：
  开机后长按*Esc*键进入GRUB引导页面（不能进入换*Shift*试试），选择*advansced options for ubuntu*，按下**e**键进入编辑界面，在**ash $vt_handoff**之间加入**nomodeset**变成**ash nomodeset $vt_handoff**。然后*Ctrl+X*应该就会正常开机了。
- 开机后：

> 开机后记得修改grub配置文件，不然每次进入都得编辑grub选项

1. 打开grub配置文件
   `sudo gedit /etc/default/grub`
2. 修改grub配置文件
   将 `GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"` 改为
   `GRUB_CMDLINE_LINUX_DEFAULT="quiet splash nomodeset"`
3. 更新grub
   `sudo update-grub`

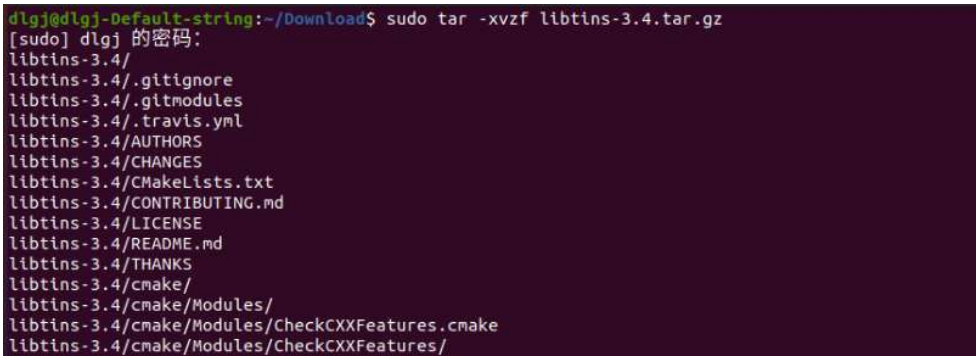
https://www.cnblogs.com/real010/p/14678546.html

2、由于某些原因而不得不使用其他更高版本 Ubuntu 系统时，在启动核心网时会出现一条名为：./bin/5gc-upfd: error while loading shared libraries: libtins.so.3.4: cannot open shared object file: No such file or directory 的报错。



究其原因，新版本的 Ubuntu 的 libtins 库默认是 4.0 版本，而核心网要求的是 3.4 版本。导致在执行脚本时找不到名为 libtins.so.3.4 的库文件。解决思路也很简单，只需要替换下库文件即可。文件地址： https://github.com/mfontanini/libtins/releases/tag/v3.4

（1）将压缩包放在任意目录下解压：



（2）进入解压完成后的 libtins-3.4 的目录后，执行以下 5 条命令

Mkdir bulid

Cd buile

Sudo cmake ..

Sudo make

Sudo make install

用来编译安装我们解压出来的库文件

```
dlgj@dlgj-Default-string:~/Download/libtins-3.4/build$ sudo make install
[100%] Built target tins
Install the project...
-- Install configuration: "RelWithDebInfo"
-- Installing: /usr/local/lib/pkgconfig/libtins.pc
-- Installing: /usr/local/CMake/libtinsConfig.cmake
-- Installing: /usr/local/CMake/libtinsConfigVersion.cmake
-- Installing: /usr/local/CMake/libtinsTargets.cmake
-- Installing: /usr/local/CMake/libtinsTargets-relwithdebinfo.cmake
-- Installing: /usr/local/include/tins/address_range.h
-- Installing: /usr/local/include/tins/arp.h
-- Installing: /usr/local/include/tins/bootp.h
```

（3）编译完成后使用命令 sudo find / -name “libtins.so*”来查看原始库文件（libtins.so.4.0）和我们刚编译完的库文件（libtins.so.3.4）分别在哪个路径下。

```
root@dlgj-Default-string:/usr/local/5gc# sudo find / -name "libtins.so*"
/usr/lib/libtins.so.4.0
/usr/lib/libtins.so
/usr/local/lib/libtins.so.3.4
/usr/local/lib/libtins.so
/usr/local/lib/libtins.so.4.5
find: ‘/run/user/1000/doc’: 权限不够
find: ‘/run/user/1000/gvfs’: 权限不够
/home/dlgj/.local/share/Trash/files/libtins-4.5/build/lib/libtins.so
/home/dlgj/.local/share/Trash/files/libtins-4.5/build/lib/libtins.so.4.5
/home/dlgj/Download/libtins-3.4/build/lib/libtins.so.3.4
/home/dlgj/Download/libtins-3.4/build/lib/libtins.so
```

（4）用命令：sudo rm -rf /usr/lib/libtins.so* 删掉原始库文件（4.0）,再用命令：cp /usr/local/lib/libtins* /usr/lib/ 把我们编译好的库文件（3.4）放到库目录下。

```
root@dlgj-Default-string:/usr/lib# sudo rm -rf /usr/lib/libtins.so*
root@dlgj-Default-string:/usr/lib# ls
accountsservice  dpkg                    gnupg                    linux                os-probes                sasl2                    tmpfiles.d
apg              eject                   gnupg2                   linux-boot-probes    os-release               shim                     ubiquity
apparmor         emacsen-common          gold-ld                  linux-sound-base     packagekit               shotwell                 ubuntu-advantage
apt              environment.d           groff                    locale               pcmciautils              snapd                    ubuntu-release-upgrader
aspell           evolution-data-server   grub                     lp_solve             pkgconfig                software-properties      udev
bfd-plugins      file                    grub-legacy              lsb                  pkg-config.multiarch     speech-dispatcher-modules udisks2
binfmt.d         firefox                 gst-install-plugins-helper man-db             pm-utils                 ssl                      ufw
bluetooth        firefox-addons          gvfs                     mentest86+           policykit-1              sudo                     unity-settings-daemon-1.0
brltty           firmware                hdparm                   mime                 pppd                     sysctl.d                 update-notifier
cnf-update-db    gcc                     ibus                     modprobe.d           pulse-13.99.1            syslinux                 upower
command-not-found gdm3                   init                     modules              python2.7                SYSLINUX                 valgrind
compat-ld        ghostscript             initramfs-tools          modules-load.d       python3                  syslinux-legacy          vino
console-setup    girepository-1.0        ispell                   netplan              python3.8                systemd                  X11
cpp              git-core                kernel                   networkd-dispatcher  python3.9                sysusers.d               x86_64-linux-gnu
crda             gnome-initial-setup     klibc                    NetworkManager       recovery-mode            tc                       xorg
cups             gnome-session           klibc-zqGeWFMIYUjZUbUCGjBo3LnHoMg.so nvidia       rhythmbox                terminfo                 xserver-xorg-video-intel
dbus-1.0         gnome-settings-daemon-3.0 language-selector      openssh              rsyslog                  thunderbird
debug            gnome-shell             libreoffice              os-prober            rygel                    thunderbird-addons
root@dlgj-Default-string:/usr/lib# sudo ldconfig
root@dlgj-Default-string:/usr/lib# cp /usr/local/lib/libtins.so* /usr/lib/
root@dlgj-Default-string:/usr/lib# ls
accountsservice  eject                   gold-ld                  linux                os-release               shotwell                 ubuntu-release-upgrader
apg              emacsen-common          groff                    linux-boot-probes    packagekit               snapd                    udev
apparmor         environment.d           grub                     linux-sound-base     pcmciautils              software-properties      udisks2
apt              evolution-data-server   grub-legacy              locale               pkgconfig                speech-dispatcher-modules ufw
aspell           file                    gst-install-plugins-helper lp_solve           pkg-config.multiarch     ssl                      unity-settings-daemon-1.0
bfd-plugins      firefox                 gvfs                     lsb                  pm-utils                 sudo                     update-notifier
binfmt.d         firefox-addons          hdparm                   man-db               policykit-1              sysctl.d                 upower
bluetooth        firmware                ibus                     mentest86+           pppd                     syslinux                 valgrind
brltty           gcc                     init                     mime                 pulse-13.99.1            SYSLINUX                 vino
cnf-update-db    gdm3                    initramfs-tools          modprobe.d           python2.7                syslinux-legacy          X11
command-not-found ghostscript            ispell                   modules              python3                  systemd                  x86_64-linux-gnu
compat-ld        girepository-1.0        kernel                   modules-load.d       python3.8                sysusers.d               xorg
console-setup    git-core                klibc                    netplan              python3.9                tc                       xserver-xorg-video-intel
cpp              gnome-initial-setup     klibc-zqGeWFMIYUjZUbUCGjBo3LnHoMg.so networkd-dispatcher recovery-mode            terminfo
crda             gnome-session           language-selector        NetworkManager       rhythmbox                thunderbird
cups             gnome-settings-daemon-3.0 libreoffice            nvidia               rsyslog                  thunderbird-addons
dbus-1.0         gnome-shell             libtins.so               openssh              rygel                    tmpfiles.d
debug            gnupg                   libtins.so.3.4           os-prober            sasl2                    ubiquity
```

（5）最后启用核心网成功

```
10/16 10:36:10.388: [sctp] INFO: MME initialize...done (../src/mme/app-init.c:33)
10/16 10:36:10.399: [app] INFO: HSS initialize...done (../src/hss/app-init.c:31)
10/16 10:36:10.400: [diam] INFO: CONNECTED TO 'hss.localdomain' (SCTP,soc#7): (../lib/diameter/common/logger.c:108)
10/16 10:36:10.400: [diam] INFO: CONNECTED TO 'mme.localdomain' (SCTP,soc#15): (../lib/diameter/common/logger.c:108)
10/16 10:36:10.451: [gtp] INFO: gtp_server() [127.0.0.4]:2123 (../lib/gtp/path.c:31)
10/16 10:36:10.452: [gtp] INFO: gtp_server() [::1]:2123 (../lib/gtp/path.c:31)
10/16 10:36:10.452: [gtp] INFO: gtp_server() [127.0.0.4]:2152 (../lib/gtp/path.c:31)
10/16 10:36:10.452: [gtp] INFO: gtp_server() [::1]:2152 (../lib/gtp/path.c:31)
10/16 10:36:10.452: [pfcp] INFO: pfcp_server() [127.0.0.4]:8805 (../lib/pfcp/path.c:31)
10/16 10:36:10.452: [pfcp] INFO: pfcp_server() [::1]:8805 (../lib/pfcp/path.c:31)
10/16 10:36:10.452: [pfcp] INFO: ogs_pfcp_connect() [127.0.0.7]:8805 (../lib/pfcp/path.c:60)
10/16 10:36:10.452: [pfcp] INFO: ogs_pfcp_connect() [127.0.0.4]:8805 (../lib/pfcp/path.c:60)
10/16 10:36:10.452: [upf] INFO: PFCP associated (../src/upf/pfcp-sm.c:173)
10/16 10:36:10.452: [sbi] INFO: nghttp2_server() [127.0.0.4]:7777 (../lib/sbi/nghttp2-server.c:145)
10/16 10:36:10.452: [app] INFO: SMF initialize...done (../src/smf/app.c:31)
10/16 10:36:10.452: [diam] INFO: CONNECTED TO 'smf.localdomain' (SCTP,soc#11): (../lib/diameter/common/logger.c:108)
10/16 10:36:10.452: [smf] INFO: PFCP associated (../src/smf/pfcp-sm.c:174)
10/16 10:36:10.453: [diam] INFO: CONNECTED TO 'pcrf.localdomain' (SCTP,soc#17): (../lib/diameter/common/logger.c:108)
10/16 10:36:10.453: [nrf] INFO: [675a4ae2-8b67-41ef-907b-47069bbe8c29] NF registred [Heartbeat:10s] (../src/nrf/nf-sm.c:193)
10/16 10:36:10.453: [smf] INFO: [675a4ae2-8b67-41ef-907b-47069bbe8c29] NF registred [Heartbeat:10s] (../src/smf/nf-sm.c:200)
10/16 10:36:12.796: [pfcp] INFO: ogs_pfcp_connect() [127.0.0.3]:8805 (../lib/pfcp/path.c:60)
10/16 10:36:12.796: [sgwu] INFO: PFCP associated (../src/sgwu/pfcp-sm.c:168)
10/16 10:36:12.796: [sgwc] INFO: PFCP associated (../src/sgwc/pfcp-sm.c:172)
*** Check ogstun: ogstun:
*** Check 5gc thread: 15
*** Success to run 5gc
```