

Manual de configurare EP06-E si 3proxy

▼ 1. Configuram tunelul VPN

Urmam cu atentie absolut fiecare pas pentru server si pentru client

https://drive.google.com/file/d/1XxAft_EwCBa7pWHZyuejUBEzs1n1Afhv/view

Adaugam la SERVER-ul nostru posibilitatea de a ne conecta prin ssh la PC-ul nostru local:

```
sudo iptables -t nat -A PREROUTING -d [IP Адрес промежуточно  
ого сервера] -p tcp --dport 2244 -j DNAT --to-dest 10.8.0.  
6:22
```

Adaugam la fel port forwarding pentru portul 13000:

```
sudo iptables -t nat -A PREROUTING -d [IP Адрес промежуточно  
ого сервера] -p tcp --dport 13000 -j DNAT --to-dest 10.8.0.  
6:13000
```

▼ 2. Configurarea modemului

```
apt update && apt upgrade -y
```

https://github.com/Sergey1560/fb4s_howto/blob/master/quectel/readme.md

Unica ce trebuie sa facem este sa introducem aceasta comanda:

```
nmcli connection add type gsm ifname '*' con-name 'MOLDCELL' apn 'm.internet'  
connection.autoconnect yes
```

Modificam "MOLDCELL" cu reseaua voastra si apn "internet" la fel. Putem sa le aflam din telefon in setari la sim-card

Daca nu ati reusit sa il configurati in Windows, vom face in Linux:

```
apt install busybox microcom - pentru trimiterea comenzilor AT
```

Apoi il transferam in modul de lucru ECM pentru a putea primi adresa DHCP

```
AT+QCFG="usbnet",1
```

Schimbam adresa lui DHCP:

`AT+QMAP="LANIP",192.168.10.1,192.168.10.2,192.168.10.2` - schimbam '10' la fiecare in parte dupa numerotatie

Adaugator dar nustiu daca neparat: `AT+QCFG="nwscanmode",0,1`

▼ Pentru a trimite comenzi AT utilizam aceasta comanda

`echo -e "ATI\r\n" > /dev/ttyUSB3 | cat /dev/ttyUSB3` - modificam locul evidentiat cu comanda dorita. Se poate si fara comanda `cat`, inasa asa vedem ce raspunde modemul.

▼ Comenzile AT

AT+GSN Посмотреть IMEI

AT+EGMR=1,7,"xxxx" Смена IMEI, где xxxxx - новый IMEI

AT+QCFG="nwscanmode" < конфигурация сети

AT+QCFG="nwscanmode",0,1 < Auto - pentru conectarea la int

AT+QCFG="nwscanmode",1,1 < GSM only

AT+QCFG="nwscanmode",2,1 < WCDMA only

AT+QCFG="nwscanmode",3,1 < 4G-LTE only

AT+QENG="servingcell" < информация о сети(ID, RSRP, RSRQ,

AT+QNWINFO < Текущий диапазон

AT+QCFG="band" < Текущий band диапазон

AT+QCAINFO < Проверка агрегации

Сброс сети на по умолчанию EP06-E:

AT+QCFG="band",8d0,1a1880800d5,0

Смена DHCP < AT+QMAP="LANIP",192.168.x.x,192.168.x.x,192.1

Настройка частот LTE:

Band значения

1 - LTE BC1
2 - LTE BC2
4 - LTE BC3
8 - LTE BC4
10 - LTE BC5
20 - LTE BC6
40 - LTE BC7

Например установить B3+B7:

Значение B3=4, значение B7=40, следовательно общее 44

AT+QCFG="band",0,44,0,1 <Первое значение - это конфиг 3G(0

AT+QCFG="band",0,44,0,1 -установить b7+B3

AT+QCFG="band",0,2000000000,0,1 -установить b38

AT+QCFG="band",0,8000000000,0,1 -установить b40

AT+QCFG="band",0,45,0,1 -установить b1+b3+b7

AT+QCFG="band",0,40,0,1 -установить b7

AT+QCFG="band",0,5,0,1 -установить b1+b3

AT+QCFG="band",0,4,0,1 -установить b3

AT+QCFG="band",0,80000,0,1 - установить b20

Тип подключения модуля

AT+QCFG="usbnet",0 - QMI/PPP/Default

AT+QCFG="usbnet",1 - ECM

AT+QCFG="usbnet",2 - MBIM

Сброс настроек на по умолчанию

AT&F

AT&F1

77887788

Список АТ команд EP06-E.txt

▼ 3. Configuram proxy-ul

In general putem sa urmam aceasta instructiune, doar ca ea se bazeaza pe modemuri Huawei si in unele momente nu este actuala si unii pasi sunt inutili, din aceasta cauza voi scrie detaliat pentru EP06-E

▼ Instalam ssh pentru a avea acces la sistema si la proxyuri de oriunde

```
apt install ssh
```

```
sudo passwd root su root
```

```
nano /etc/ssh/sshd_config
```

unde este `port 22` si `PermitRootLogin` stergem dezcomentam si modificam la `PermitRootLogin yes` apoi apasam CTRL+X, y si enter - pentru a salva

```
service sshd restart service ssh restart
```

▼ 1. Configurarea retelei

```
ip a nano /etc/default/grub
```

```
GRUB_CMDLINE_LINUX="net.ifnames=0 biosdevname=0" update-grub
```

```
apt install ifupdown
```

```
cd /etc/network/ nano interfaces
```

```
auto eth0
iface eth0 inet static
    address 192.168.0.110
    netmask 255.255.255.0
    gateway 192.168.0.1
    dns-nameservers 192.168.0.1
auto usb0 usb1
iface usb0 inet dhcp
iface usb1 inet dhcp
```

adaugam atitea cate modemuri avem

```
reboot ip a
```

verificam daca totul e bine configurat prin `systemctl restart networking`

▼ 2. Configurarea retelei

```
nano /etc/iproute2/rt_tables
```

```
100 wan
101 T1
102 T2
```

adaugam atitea cate modemuri avem

`ifconfig` ne uitam ce interfata avem la internet - ex. **eth0** sau **ens3**

```
nano /etc/network/if-up.d/routes-eth0
```

 - ar trebui sa fie `eth0`

```
#!/bin/sh
if [ "$IFACE" = "eth0" ]; then
    ip route add 192.168.0.0 dev eth0 src 192.168.0.110 ta
    ip route add default via 192.168.0.1 table wan
    ip route add 192.168.0.0 dev eth0 src 192.168.0.110
    ip rule add from 192.168.0.110 table wan
fi
```

modificam a cu adresa evidentiata cu a noastra locala

```
nano /etc/network/if-up.d/routes-usb0
```

```
#!/bin/sh
if [ "$IFACE" = "usb0" ]; then
    ip route add 192.168.10.0 dev usb0 src 192.168.10.2 ta
    ip route add default via 192.168.10.1 table T1
    ip route add 192.168.10.0 dev usb0 src 192.168.10.2
    ip rule add from 192.168.10.2 table T1
fi
```

facem la fel pentru fiecare modem cu modificarea informatiei evidentiata

```
chmod +x /etc/network/if-up.d/routes-eth0
```

```
chmod +x /etc/network/if-up.d/routes-usb0
```

```
reboot
```

Putem verifica prin comanda `ping 192.168.10.2` si `ping 192.168.10.2 google.com`

▼ Configurarea 3proxy

```
apt-get -y install gcc g++ git make curl
```

Copiem totul si punem in terminal

```
cd ~
wget --no-check-certificate https://github.com/z3APA3A/3proxy
tar xzf 0.8.12.tar.gz
mv ~/3proxy-0.8.12 ~/3proxy
cd ~/3proxy
make -f Makefile.Linux
```

Aici la fel

```
mkdir /usr/local/3proxy
cd ~/3proxy/src
cp 3proxy /usr/local/3proxy/
cd /etc/init.d/
wget http://kak-podnyat-proksi-ipv6.ru/skript/3proxy
chmod +x /etc/init.d/3proxy
update-rc.d 3proxy defaults
cd /usr/local/3proxy/
nano /usr/local/3proxy/3proxy.cfg
```

Se deschide file-ul cu configurarea lui 3proxy

```
monitor /usr/local/3proxy/3proxy.cfg

daemon
timeouts 1 5 30 60 180 1800 15 60
maxconn 5000
nscache 65535
log /dev/null
```

```

auth strong
users login:CL:pass
allow login
proxy -n -a -p13000 -i10.8.0.6 -e192.168.10.2
proxy -n -a -p13001 -i10.8.0.6 -e192.168.11.2
proxy -n -a -p13002 -i10.8.0.6 -e192.168.12.2
flush

```

adaugam cate modeme le trebuie si cu ce port ne trebuie

`-p13000` - portul

`-i10.8.0.6` - adresa tunelului VPN

`-e192.168.10.2` - adresa modemului locala dupa DHCP, o putem vedea in `ifconfig`

`users login:CL:pass` - modificam cu login-ul si parola dorita, apoi modificam si aceasta cu loginul ales `allow login`

mai detaliat despre 3proxy - <https://bozza.ru/art-94.html>

`/etc/init.d/3proxy start`

`reboot`

`nano /etc/sysctl.conf`

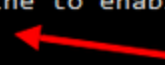
```

# Uncomment the next two lines to enable Spoof protection (reve
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1
# Uncomment the next line to enable packet forwarding for IPv6

```



Dezcomentam si adaugam la sfarsit de file:

```
net.ipv4.route.min_adv_mss = 256
net.ipv4.tcp_rmem = 8192 87380 16777216
net.ipv4.tcp_wmem = 6144 87380 1048576
net.ipv4.icmp_echo_ignore_all = 1
```

```
sysctl -p
```

▼ Port forwarding

Permite port forwarding-ul pentru PC-ul nostru si deschidem portul `13000`

```
# INPUT: Allow incoming traffic to the forwarded port
sudo iptables -I INPUT -p tcp --dport 13000 -m conntrack -

# FORWARD: Allow forwarded traffic
sudo iptables -I FORWARD -p tcp --dport 13000 -m conntrack

# POSTROUTING: Allow response traffic out (assuming you're
sudo iptables -t nat -I POSTROUTING -o tun0 -j MASQUERADE

# Allow related and established traffic back out
sudo iptables -I OUTPUT -p tcp --sport 13000 -m conntrack
```

Aici trebuie neaparat de scris interfata noastra de la tunelul VPN - `tun0`

```
sudo iptables-save
```

```
sudo iptables-save > /etc/iptables.conf
```

```
sudo nano /etc/network/if-up.d/rules_ipv4
```

```
#!/bin/bash
iptables-restore /etc/iptables.conf
```

```
sudo chmod +x /etc/network/if-up.d/rules_ipv4
```

Verificam - `nmap` `<adresa server-ului>` `-p 13000` si `ip route`

Daca nu aveti nmap - `apt install nmap`

▼ Reconectarea modemelor

Pentru a schimba IP-ul modemelor fiecare 2 min, trebuie de creat un script in bash care va pune modemul in mod avion si il va reseta apoi pentru a avea iarasi internet. La trecerea modemului in mod avion, se schimba IP-ul modemului.

`cd /root` si creem aici un fisier cu codul pentru reconect.

`nano reconnect.sh`

```
#!/bin/bash

echo -e "AT+CFUN=4\r\n" > /dev/ttyUSB3
sleep 1
echo -e "\e[32mSuccessfully in airplane mode\e[0m"

echo -e "AT+CFUN=1\r\n" > /dev/ttyUSB3
sleep 1
echo -e "AT+CREG?\r\n" > /dev/ttyUSB3
echo -e "\e[32mSuccessfully changed the IP address\e[0m"
```

`chmod +x reconnect.sh`

Apoi adaugam in cron acest fisier

`crontab -e`

- `/2 * * * * /root/reconnect.sh`

▼ Conectarea la proxy

Modificati cu ale voastre

adresa statica - 193.36.38.24

loghin - admin

password - pass

port - 13000