VPN!

Πίνακας περιεχομένων

1. Install swarmlab-sec (Home PC)

HowTo: See http://docs.swarmlab.io/lab/sec/sec.adoc.html

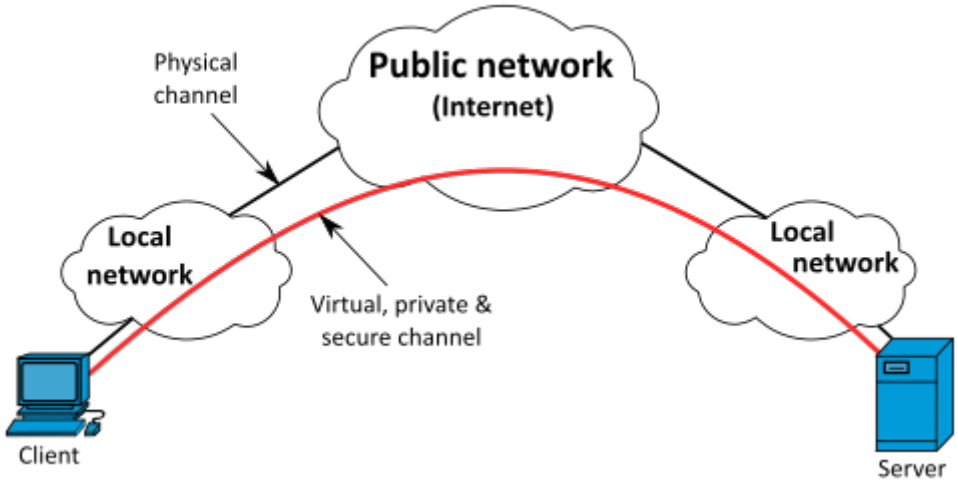
NOTE

Assuming you're already logged in

2. VPN

A **virtual private network (VPN)** extends a private network across a public network, and enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network. Applications running on a computing device, e.g., a laptop, desktop, smartphone, across a VPN may therefore benefit from the functionality, security, and management of the private network. Encryption is a common, though not an inherent, part of a VPN connection

More: wikipedia



NOTE

OpenVPN is an open-source software that implements virtual private network (VPN) techniques to create secure point-to-point or site-to-site connections in routed or bridged configurations and remote access facilities. It uses a custom security protocol that utilizes SSL/TLS for key exchange. It is capable of traversing network address translators (NATs) and firewalls. It was written by James Yonan and is published under the GNU General Public License (GPL).

More: wikipedia

3. Create VPN

create-vpn.sh

```
#!/bin/bash
IP=192.168.89.5                                          # Server IP
P=1194                                                   # Server Port
OVPN_SERVER='10.80.0.0/16'                               # VPN Network
vpn_data=/var/lib/swarmlab/openvpn/openvpn-services/     # Dir to save data ** this
must exist **
NAME=swarmlab-vpn-services                               # name of docker service
DOCKERnetwork=swarmlab-vpn-services-network              # docker network
docker=registry.vlabs.uniwa.gr:5080/myownvpn             # docker image

docker stop  $NAME                  #stop container
sleep 3
docker container rm  $NAME              #rm container

# rm config files
sudo rm -f $vpn_data/openvpn.conf.*.bak
sudo rm -f $vpn_data/openvpn.conf
sudo rm -f $vpn_data/ovpn_env.sh.*.bak
sudo rm -f $vpn_data/ovpn_env.sh

# create network
sleep 2
```

```
docker network create --attachable=true --driver=bridge --subnet=172.50.0.0/16
--gateway=172.50.0.1 $DOCKERnetwork

read -d '' MULTILINE_EXTRA_SERVER_CONF << EOF
duplicate-cn
max-clients 35000
topology subnet
EOF

#run container
sleep 3
docker run --net=none -it -v $vpn_data:/etc/openvpn  --rm $docker ovpn_genconfig  -u
udp://$IP:1194 \
-N -d -c -p "route 172.50.20.0 255.255.255.0" -e "topology subnet" -s $OVPN_SERVER


# create pki
sleep 3
echo "new pki is disabled"
docker run --net=none -v $vpn_data:/etc/openvpn  --rm -it $docker ovpn_initpki

#sleep 3
#docker run --net=none -v $vpn_data:/etc/openvpn  --rm $docker ovpn_copy_server_files

#create vpn
sleep 3
docker run --detach --name $NAME -v $vpn_data:/etc/openvpn --net=$DOCKERnetwork
--ip=172.50.0.2 -p $P:1194/udp --cap-add=NET_ADMIN $docker

sleep 5
sudo sysctl -w net.ipv4.ip_forward=1

#show created
docker ps
```

4. Create user

config

```
#!/bin/bash
IP=83.212.114.14
P=5194
vpn_data=/var/lib/swarmlab/openvpn/openvpn-services/
NAME=swarmlab-vpn-services
DOCKERnetwork=swarmlab-vpn-services-network
docker=registry.vlabs.uniwa.gr:5080/myownvpn
PATHNAME=/var/lib/swarmlab/openvpn/etc/vpn-data_user_config
vpn_data_user_config=$PATHNAME

vpn_data=/var/lib/swarmlab/openvpn/openvpn-services/
vpn_data_user_config=/var/lib/swarmlab/openvpn/etc/vpn-data_user_config
NAME=swarmlab-vpn-services

MANAGER=/var/lib/swarmlab/openvpn/etc/managers
WORKER=/var/lib/swarmlab/openvpn/etc/workers
MANAGERkeys=/var/lib/swarmlab/openvpn/etc/managers_keys
```

create-user.sh

```
#!/bin/bash

. ./config

sudo mkdir -p $vpn_data
sudo mkdir -p $vpn_data_user_config
sudo mkdir -p $MANAGERkeys

docker=registry.vlabs.uniwa.gr:5080/myownvpn
echo $vpnip
echo $#

docker=registry.vlabs.uniwa.gr:5080/myownvpn
echo $vpnip
echo $#

if [ $# -eq 1  ]; then
        CLIENTNAME=$1
        U=$CLIENTNAME
        mkdir users
        docker run -v $vpn_data:/etc/openvpn --rm -it $docker easyrsa build-client-
full $CLIENTNAME nopass
        sleep 3
        docker run -v $vpn_data:/etc/openvpn --log-driver=none --rm $docker
ovpn_getclient $CLIENTNAME  > users/$CLIENTNAME.ovpn

        file="users/$CLIENTNAME.ovpn"
```

```
        ps='remote '
        pi="remote $IP $P udp"
        grep -q "^$ps" $file && sed -i "s/^$ps.*/$pi/" $file || sed -i "5a $pi" $file

        ps='comp-lzo'
        pi='comp-lzo no'
        grep -q "^$ps" $file && sed -i "s/^$ps.*/$pi/" $file || sed -i "6a $pi" $file

        ps='resolv-retry'
        pi='resolv-retry infinite'
        grep -q "^$ps" $file && sed -i "s/^$ps.*/$pi/" $file || sed -i "7a $pi" $file
        ps='persist-key'
        pi='persist-key'
        grep -q "^$ps" $file && sed -i "s/^$ps.*/$pi/" $file || sed -i "8a $pi" $file

        ps='persist-tun'
        pi='persist-tun'
        grep -q "^$ps" $file && sed -i "s/^$ps.*/$pi/" $file || sed -i "9a $pi" $file

        ps='keepalive'
        pi='keepalive 15 60'
        grep -q "^$ps" $file && sed -i "s/^$ps.*/$pi/" $file || sed -i "10a $pi" $file



else
        echo "no clientname"
fi
```

5. rm vpn user

rm-user.sh

```
#!/bin/bash
. ./config

CLIENTNAME=$1
U=$CLIENTNAME

if [ $# -eq 1  ]; then
        sudo rm -f $vpn_data/pki/reqs/$CLIENTNAME.req
        sudo rm -f $vpn_data/pki/private/$CLIENTNAME.key
        sudo rm -f $vpn_data/pki/issued/$CLIENTNAME.crt
        sudo rm -f $vpn_data/server/ccd/$CLIENTNAME
        sudo rm -f $vpn_data/ccd/$CLIENTNAME
        pem=$(sudo grep "CN=$U$"  $vpn_data/pki/index.txt | cut  -f4)
        #/var/lab/gswarm/vpn-
data/pki/certs_by_serial/BACA61827E65D0E5F695245519410952.pem
        sudo rm -f $vpn_data/pki/certs_by_serial/$pem.pem
        sudo sed -i "/CN=$U$/d"  $vpn_data/pki/index.txt
        echo $pem
        docker run -v $vpn_data:/etc/openvpn --log-driver=none --rm -it $docker
ovpn_revokeclient  $CLIENTNAME remove


        sudo rm -f $vpn_data_user_config/$CLIENTNAME.ovpn
        sudo rm -f $vpn_data_user_config1/$CLIENTNAME.ovpn
else
        echo "no client"
fi
```

6. show all vpn users

show-user.sh

```
. ./config

docker exec -it  $NAME ovpn_listclients
```

7. show all connected vpn users

show-conn-user.sh

```
. ./config

docker exec -it  $NAME  cat /tmp/openvpn-status.log
```

Reminder

Caminante, no hay camino,
se hace camino al andar.

Wanderer, there is no path,
the path is made by walking.

Antonio Machado Campos de Castilla