# OpenVPN Server

Tento mini návod si dává za úkol popsat instalaci OpenVPN server na vpsfree za použití autentizace RSA klíči (vygenerovanými easy-rsa scripty) při použití distribuce Ubuntu 10.04 nebo Debian 6.

V tomto článku používám testovací název serveru **torm**, doménu **example.com** a uživatele **arteal** - tyto hodnoty si ve své instalaci prosím upravte.

Tento článek nepoužívá k zabezpečení klíčů žádné heslo, pokud se necítíte, že vaše certifikáty budou ve vašem počítači dostatečně v bezpečí, můžete si je zabezpečit heslem a openvpn se vás při připojování na toto heslo zeptá.

## Nastavení ve vpsadminu

V nastavení naší VPS zapneme potřebné vlastnosti (TUN/TAP driver) :

VPS→(ID VPS)→Enable Features→Enable All→Go »

| Enable features | |
|---|---|
| Enable TUN/TAP | disabled |
| Enable iptables | disabled |
| Enable FUSE | disabled |
| NFS server + client | disabled |
| PPP | disabled |
| Enable all: | ☐ |
| | Go >> |

## Potřebné balíky

Jelikož jsou vpsfree templaty distribucí minimalistické nainstalujeme kromě openvpn i některé další balíky :

```
root@torm:~# apt-get install rsyslog dialog apt-utils easy-rsa openvpn
```

(je možné, že pro správnou instalaci je nejdříve nutné spustit *apt-get update*)

## nastavení systemd ve VPSadminOS

Pokud používáte např. Debian 9, Fedoru, openSUSE, či jinou distribuci se systemd, musíte nejprve aktivovat patřičnou službu - pro konfiguraci v */etc/openvpn/server/server.conf* spusťte

```
root@torm:~# systemctl enable openvpn-server@server.service
```

A následovně ještě oeditujte příslušnou službu:

```
root@torm:~# systemctl enable openvpn-server@server.service
[Service]
LimitNPROC=infinity
```

Poté již můžete službu nastartovat:

```
root@torm:~# systemctl start openvpn-server@server.service
```

# Vygenerování certifikátů

Pro použití jednoduchého CA použijeme EasyRSA, které je „přiložené“ k OpenVPN :

```
root@torm:~# cp -R /usr/share/easy-rsa /etc/openvpn/
root@torm:~# cd /etc/openvpn/easy-rsa

Nyní upravíme soubor (/etc/openvpn/easy-rsa)vars, ve kterém je dobré upravit
miniálně posledních 5 řádků:

export KEY_COUNTRY="CZ"
export KEY_PROVINCE="CZ"
export KEY_CITY="Brno"
export KEY_ORG="ExampleCompany"
export KEY_EMAIL="admin@example.com"
```

Tyto hodnoty načteme :

```
root@torm:/etc/openvpn/easy-rsa# source ./vars
```

Vytvoříme složku pro certifikáty :

```
root@torm:/etc/openvpn/easy-rsa# ./clean-all
```

Vygenerujeme DH certifikát:

```
root@torm:/etc/openvpn/easy-rsa# ./build-dh
Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
.................................+.....................................+...+
.........................++*++*++*
```

Vygenerujeme CA ceritifikát :

```
root@torm:/etc/openvpn/easy-rsa# ./build-ca
```

[Zde neuvádím plný výstup, pokud jste správně upravily soubor vars, lze hodnoty ponechat na výchozním nastavení - pouze „odentrovat“]

Vygenerujeme certifikát pro server (v mém příkladu server jménem „torm“) :

```
root@torm:/etc/openvpn/easy-rsa# ./build-key-server torm
```

[Zde neuvádím plný výstup, pokud jste správně upravily soubor vars, lze hodnoty ponechat na výchozním nastavení - pouze „odentrovat“, pouze potvrdit, že chceme daný certifikát podepsat. Dále nepočítám, že certifikát chceme šifrovat heslem]

Vygenerujeme uživatelský certifikát :

```
root@torm:/etc/openvpn/easy-rsa# ./build-key arteal
```

[Zde neuvádím plný výstup, pokud jste správně upravily soubor vars, lze hodnoty ponechat na výchozním nastavení - pouze „odentrovat“, pouze potvrdit, že chceme daný certifikát podepsat. Dále nepočítám, že certifikát chceme šifrovat heslem]

# Nastavení OpenVPN serveru

/etc/openvpn/torm.conf

```
mode server
tls-server
port 1194
proto tcp-server
dev tap1
client-config-dir ccd

ca /etc/openvpn/easy-rsa/keys/ca.crt
cert /etc/openvpn/easy-rsa/keys/torm.crt
key /etc/openvpn/easy-rsa/keys/torm.key
dh /etc/openvpn/easy-rsa/keys/dh1024.pem

ifconfig 10.50.10.254 255.255.255.0
ifconfig-pool 10.50.10.1 10.50.10.20 255.255.255.0
ifconfig-pool-persist ipp.txt

client-to-client
#route 10.20.15.0 255.255.255.0 10.50.10.254
#push "route 10.20.15.0 255.255.255.0 10.50.10.254 3"

keepalive 10 120
comp-lzo
user nobody
group nogroup
persist-key
persist-tun
```

```
status /tmp/openvpn.status 1
log-append /var/log/openvpn.log
status-version 3
verb 4
mute 20
```

Tento příklad využívá TCP protokolu, certifikátů, které jsme si vygenerovali. Používáme klientský IP rozsah 10.50.10.1-10.50.10.20, IP jsou persistentně ukládány do souboru /etc/opevpn/ipp.txt, ale zároveň lze využít ccd(Client Config Dir). VPN dovoluje komunikaci mezi klienty, dále vidíme zakomentovanou route do sítě 10.20.15.0/24, přes náš VPN server. Server udržuje spojení odesílání TCP packetu každých 10 sekund a pokusí se restartovat spojení, pokud nedostane ping po 120 sekund. Využíváme LZO komprese, spouštíme pod neprivilegovaným uživatelem z bezpečnostních důvodů, při restartu nezahazujeme klíče ani síťové rozhraní. Aktuální stav připojených klientů najdeme v souboru /tmp/openvpn.status a log v souboru log-append /var/log/openvpn.log

Nyní můžeme spustit OpenVPN server :

```
root@torm:/etc/openvpn# /etc/init.d/openvpn start
* Starting virtual private network daemon(s)...
*   Autostarting VPN 'torm'
[ OK ]
```

# Nastavení OpenVPN klienta :

Klientovi předáme tyto soubory (ze složky /etc/openvpn/easy-rsa/2.0/keys) :

- ca.crt
- arteal.crt
- arteal.key

torm.conf

```
client
remote torm.example.com 1194
ca ca.crt
cert arteal.crt
key arteal.key
comp-lzo yes
dev tap5
proto tcp
nobind
auth-nocache
script-security 2
persist-key
persist-tun
pull
log-append /var/log/openvpn.log
```

Na MS Windows se tento soubor musí jmenovat torm.ovpn a direktiva log-append nebude fungovat, je lepší ji tedy zakomentovat.

# Routování veškerého trafficu přes OpenVPN

Jelikož jsem slyšel, že jsou požadavky na používání OpenVPN jako gateway, tedy na směrování veškerého přenosu přes VPN, rozhodl jsem se tento článek rošířit o část, která právě toto řeší.

## Úprava nastavení

/etc/openvpn/torm.conf

```
...
client-to-client
#route 10.20.15.0 255.255.255.0 10.50.10.254
#push "route 10.20.15.0 255.255.255.0 10.50.10.254 3"
push "route-gateway 10.50.10.254"
push "redirect-gateway def1"

keepalive 10 120
comp-lzo
...
```

## Nastavení VPS

```
root@torm:~#apt-get install iptables
root@torm:~#iptables -t nat -A POSTROUTING  -o venet0 -j SNAT --to x.x.x.x
```

Zde nahradíte x.x.x.x za adresu vaší VPS (kterou vyčtete mj. z `ip addr show dev venet0:0 scope global`)

# Alternativni VPN server pres Docker

Alternativou je rozchodit si Docker a poustet VPN jako predpripraveny image pro Docker. Navod ke zprovozneni na https://github.com/kylemanna/docker-openvpn