

OpenVZ Nastavení sítě

Virtual network device

Pokud nepotřebujeme žádné speciality v nastavení sítě, tj. nepotřebujeme VLANy, bridge atp. uvnitř VPS, vystačíme si se standardním výchozím nastavení OpenVZ. V tomto případě se používá výchozí zařízení venet0 a ip adresy se nastavují přímo do konfigurace vps.

[/etc/vz/1000.conf](#)

```
IP_ADDRESS="77.87.240.71/26 2001:67c:2190::10/64 10.11.56.205/26"
```

```
vzctl set 1000 --ipadd 77.87.240.71/26 --save  
vzctl set 1000 --ipadd 2001:67c:2190::10/64 --save
```

Podrobnosti naleznete zde [Virtual_network_device](#)

Virtual Ethernet device

Pokud chceme v rámci VPS používat kompletní možnosti síťování včetně nastavení VLANu, použijeme virtuální ethernet. Předností tohoto řešení je, že je možné nastavení sítě včetně firewallu realizovat standardními konfiguračními souborami a příkazy v rámci konkrétní distribuce, kterou si do VPS nainstalujeme. Správný postup používání Virtual Ethernetu je zde:

1. Zajistíme automatické načtení všech modulů, potřebných pro síťování. Např. pokud chceme v rámci VPS používat VLANy, musíme nastavit při startu hostitelského systému automatické načtení modulu 8021q. V centosu se používá soubor **/etc/rc.modules**, který je nutno vytvořit spustitelný. V debianu se používá **/etc/modules**

[/etc/rc.modules](#)

```
modprobe 8021q
```

2. Vytvoříme bridge např. **virbr0** a jeho prvním členem bude fyzický interface na routeru, např. **eth1**



Pokud plánujete na serveru používat současně Virtual network a Virtual ethernet, používejte pro obě možnosti různé fyzické interfacery - v mém případě je to eth0 pro virtual network venet0 a eth1 pro virtual ethernet

3. Vytvoříme VPS, nebudeme specifikovat zadné IP adresy ani rozhraní. Po vytvoření VPS použijeme tento příkaz pro založení virtual ethernet

```
vzctl set 1000 --netif_add eth0,,,virbr0 --save
```

Po nastartovani VPS vznikne dalsi interface veth1000

4. Nyní je potřeba zajistit aby se virtualni interface stal po nastartování VPS automaticky členem bridge virbr0 a sitovani fungovalo tak jak ma. Je nutne vytvořit následující konfigurační soubor:

</etc/vz/vznet.conf>

```
EXTERNAL_SCRIPT="/usr/sbin/vznetaddbr"
```

Nyní VPS nastartujeme:

```
root@aquila:/etc/vz/conf# vzctl start 1000
Starting container...
Container is mounted
Setting CPU units: 6000
Setting CPUs: 2
Configure veth devices: veth1000.1
Adding interface veth1000.1 to bridge virbr0 on CT0 for CT1000
Container start in progress...
```

5. Po nastartování VPS můžeme provést klasickou konfiguraci sítě pomocí distribuce pod VPS např. v debianu

</etc/network/interface>

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 10.11.57.60
    netmask 255.255.255.128
    gateway 10.11.57.1

auto vlan10
iface vlan10 inet static
    address 192.168.11.1
    netmask 255.255.255.0
    vlan_raw_device eth0
```

a nahodit interfaci.

Podrobnosti naleznete zde [Virtual_Ethernet_device](#)

Použití obou režimů současně

Pokud plánujete oba síťové režimy používat současně, doporučuji pro každý z nich použít jednu fyzickou síťovku na serveru. Každý server má dnes minimálně dvě síťové karty, takže já používám **eth0 pro Virtual Network** a **eth1 pro bridge určený pro jednotlivé Virtual Ethernety**

V konfiguraci OpenVZ je pak dobré specifikovat, přes které rozhraní se bude routovat venet0:

</etc/vz/vz.conf>

```
VE_ROUTE_SRC_DEV="eth0"  
NEIGHBOUR_DEVS=all
```

From:

<https://wiki.spoje.net/> - **SPOJE.NET**

Permanent link:

https://wiki.spoje.net/doku.php/howto/vps/openvz/openvz_networking

Last update: **2014/02/24 19:27**

