Subject: Re: MediaTomb UPnP server
Posted by starfry on Sun, 26 Jul 2009 10:57:07 GMT
View Forum Message <> Reply to Message

Here is how to set up veth networking woth OpenVZ. These notes relate to version 3.0.22 on Ubuntu 8.04 Hardy Heron.

The first thing you must do is create a network bridge on your host.

Install the Bridge Utilities

apt-get install bridge-utils

Modify your network configuration file /etc/network/interfaces. Here is what mine looks like:

This file describes the network interfaces available on your system
and how to activate them. For more information, see interfaces(5).

The loopback network interface
auto lo
iface lo inet loopback

The primary network interface (bridged configuration)
auto br0
iface br0 inet dhcp
    bridge_ports eth0

Reboot, or re-start networking to invoke the new configuration
#/etc/init.d/networking restart

Check you have a bridge (use ifconfig) and that your network still works before proceeding.

Next, there are some config files and scripts needed to enhance openvz so it does the required setup work to add/remove a vm from the bridge. There are three files:

    /usr/sbin/vznetcfg
    /usr/sbin/vznetaddbr
    /etc/vz/vznet.conf

The first file, /usr/sbin/vznetcfg, already exists but causes a problem on ubuntu because the default shell (which
is actually "dash") does not support the "source" command. Change line 20...

from this:
[-f "$VZNETCFG"] && source "$VZNETCFG"
to this:
[-f "$VZNETCFG"] && . "$VZNETCFG"

The second file, /ust/sbin/vznetaddbr, is a new script that is invoked to do the bridge configuration:

```
#!/bin/sh
# /usr/sbin/vznetaddbr
# a script to add virtual network interfaces (veth's) in a CT to a bridge on CT0

CONFIGFILE=/etc/vz/conf/$VEID.conf
. $CONFIGFILE
VZHOSTIF=`echo $NETIF |sed 's/^.*host_ifname=\(.*\),.*$/\1/g'`

if [ ! -n "$VZHOSTIF" ]; then
   echo "According to $CONFIGFILE CT$VEID has no veth interface configured."
   exit 0
fi

if [ ! -n "$VZHOSTBR" ]; then
   echo "According to $CONFIGFILE CT$VEID has no bridge interface configured."
   exit 1
fi

echo "Adding interface $VZHOSTIF to bridge $VZHOSTBR on CT0 for CT$VEID"
/sbin/ifconfig $VZHOSTIF 0
echo 1 > /proc/sys/net/ipv4/conf/$VZHOSTIF/proxy_arp
echo 1 > /proc/sys/net/ipv4/conf/$VZHOSTIF/forwarding
/usr/sbin/brctl addif $VZHOSTBR $VZHOSTIF

exit 0
```

The third file, /etc/vz/vznet.conf, contains a reference to the second file and is used by the first file, which is invoked during bridge setup:

```
#!/bin/sh
EXTERNAL_SCRIPT="/usr/sbin/vznetaddbr"
```

With those files in place, openvz should manage connecting your veth vps to your bridge. You just need to create the VPS and configure its networking. Here's how:

First, create the vps:

```
vzctl create 123 --ostemplate ubuntu-8.04-i386-minimal
vzctl set 123 --hostname vz123 --save
```

next, add the bridge to the vps configuration:

```
echo "CONFIG_CUSTOMIZED=\"yes\"" >> /etc/vz/conf/123.conf
echo "VZHOSTBR=\"${BRIDGE:-br0}\"" >> /etc/vz/conf/123.conf
```

and then add an eth0 device to it:

```
vzctl set 123 --netif_add eth0 --save
```

Now configure networking within the vps. This basically involves creating an appropriate /etc/network/interfaces file.

For DHCP addressing, create the /etc/network/interfaces file like this:
```
cat > /vz/private/123/etc/network/interfaces << EOF
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
EOF
```

For static addressing, create the /etc/network/interfaces file like this:
```
cat > /vz/private/123/etc/network/interfaces << EOF
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
    address 10.0.0.123
    netmask 255.0.0.0
    gateway 10.0.0.138
EOF
```

If we're doing static addressing (this is not required if using dhcp because dhcp takes care of it), create the /etc/resolv.conf file like this:
```
cat > /vz/private/123/etc/resolv.conf << EOF
search lan
nameserver 10.0.0.138
EOF
```

That's it! The VPS is now ready for use, so start it up:
```
vzctl start 123
```

You can test your setup works for multicast with a utility called ssmping. You need to install it on the VPS and also on another host (could be ct0 or a completely different machine on the network):

Install on the VPS (from the host):
```
cp /etc/apt/sources.list /vz/root/123/etc/apt
vzctl exec 123 apt-get update
vzctl exec 123 apt-get install ssmping
```

Install elsewhere:
```
apt-get install ssmping
```

To test, you run "ssmpingd" the vps and you run "ssmping" on another machine:

```
vzctl enter 123
ssmpingd
```

And on the other machine
ssmping 10.0.0.123

You should get output from ssmpingd:
root@vz123:/# ssmpingd
received request from 10.0.2.118
received request from 10.0.2.118
received request from 10.0.2.118
received request from 10.0.2.118
received request from 10.0.2.118

You should get output from ssmping:
john@tma1:~$ ssmping 10.0.0.123
ssmping joined (S,G) = (10.0.0.123,232.43.211.234)
pinging S from 10.0.2.118
  unicast from 10.0.0.123, seq=1 dist=0 time=5.609 ms
multicast from 10.0.0.123, seq=1 dist=0 time=5.621 ms
  unicast from 10.0.0.123, seq=2 dist=0 time=0.036 ms
multicast from 10.0.0.123, seq=2 dist=0 time=0.045 ms
  unicast from 10.0.0.123, seq=3 dist=0 time=0.023 ms
multicast from 10.0.0.123, seq=3 dist=0 time=0.030 ms
  unicast from 10.0.0.123, seq=4 dist=0 time=0.026 ms
multicast from 10.0.0.123, seq=4 dist=0 time=0.032 ms
  unicast from 10.0.0.123, seq=5 dist=0 time=0.025 ms
multicast from 10.0.0.123, seq=5 dist=0 time=0.033 ms

--- 10.0.0.123 ssmping statistics ---
5 packets transmitted, time 4592 ms
unicast:
   5 packets received, 0% packet loss
   rtt min/avg/max/std-dev = 0.023/1.143/5.609/2.233 ms
multicast:
   5 packets received, 0% packet loss since first mc packet (seq 1) recvd
   rtt min/avg/max/std-dev = 0.030/1.152/5.621/2.234 ms
john@tma1:~$

If you change the networking of an existing venet0 VPS you may need to remove old routes from the routing table because they conflict with the veth setup (this is not necessary on a clean VPS built according to the above instructions). This command will achieve that:

complex command greps routes with venet in them and then removes them.
IFS is used to delimit on line breaks instead of word breaks. Shell
characters are escaped to avoid interpretation on the host.
vzctl exec 123 "IFS=\`echo -en \"\\n\\b\"\` && for r in \`ip route | grep venet0\`; do eval \"ip route del \$r\"; done"

With that all done, you should be able to install Mediatomb or Mythtv and see them on your PS3. I won't go in to the details of those apps as I expect you know that already. The key is to have the network set up correctly and I hope my notes go some way towards helping you achieve that

The other thing is to make sure your PS3 has the latest firmware so that it supports Upnp, but I expect you have that already!

I have attached two example scripts containing the relevant commands.

File Attachments

1) vps_dhcp_example, downloaded 716 times
2) vps_static_example, downloaded 787 times