Subject: Giving a VPS a MAC address
Posted by Jason Stubbs on Sat, 04 Mar 2006 10:39:49 GMT
View Forum Message <> Reply to Message

In situations where a VPS needs a unique MAC address this guide can be used to give it one. The method used will be to create a tunneled bridge between the hardware node and the vps.

Note, this method causes further processing overhead so only add one if you need to. Also worth mentioning is that this will bypass all security that is offered by the venet virtual device. Lastly, each VPS needing a MAC address will require a dedicated tap device on the hardware node tied to the bridge.

In this guide, the local network is 192.168.0.0/24, the "internal" network (for communication between the hardware node and the VPS) is 192.168.1.0/24, the hardware node's IP will be 192.168.0.1, the VPS's IP will be 192.168.0.2 and the gateway is 192.168.0.253.

The software needed:

* kernel compiled with bridging and tun/tap support
* vtun (vtun.sf.net)
* bridge-utils (bridge.sf.net)
* usermode-utilities (user-mode-linux.sf.net)

HARDWARE NODE

Create a tap device, bridge it with your ethernet device and set the IPs on the bridge:

```
# tunctl -t tap0
# ifconfig eth0 0.0.0.0 promisc up
# ifconfig tap0 0.0.0.0 promisc up
# brctl addbr br0
# brctl addif br0 eth0
# brctl addif br0 tap0
# ifconfig br0 192.168.0.1 up
# ifconfig br0:1 192.168.1.1 up
```

Then set up vtun to run as a server:

```
# cat /etc/vtund-start.conf
--server-- 5000

# cat /etc/vtund.conf
options {
  port 5000;
  syslog       daemon;
  ppp          /usr/sbin/pppd;
```

```
  ifconfig      /sbin/ifconfig;
  route         /sbin/route;
  firewall      /sbin/ipchains;
  ip            /sbin/ip;
}
vps101 {
  speed 0;
  compress no;
  encrypt no;
  passwd  <passwd>;
  type ether;
  device tap0;
  proto udp;  # perhaps tcp is faster? haven't tested yet...
}

# /etc/init.d/vtund start
```

VPS 101

You will need to allow access to tun devices by the VPS and give it an IP on the "internal" network.

```
# vzctl set 101 --devices c:10:200:rw --save
# vzctl set 101 --ipadd 192.168.1.2 --save
```

Then just set up and run vtund within the VPS:

```
# cat /etc/vtund-start.conf
vps101 192.168.1.1

# cat /etc/vtund.conf
options {
  port 5000;
  timeout 60;
  ppp           /usr/sbin/pppd;
  ifconfig      /sbin/ifconfig;
  route         /sbin/route;
  firewall      /sbin/ipchains;
  ip            /sbin/ip;
}
vps101 {
  passwd <passwd>;
  type ether;
  device tap1;
  up {
    ifconfig "%% 192.168.0.2 netmask 255.255.255.0";
  };
```

```
  down {
    ifconfig "%% down";
  };
}
```

```
# /etc/init.d/vtund start
```

Finally, you'll need to adjust routing on the VPS to access the gateway directly or else the "internal" address will be used when sending packets outside the local network and the gateway won't know how to send responses back.

```
# route del default
# route add -net 192.168.1.0/24 dev venet0
# route add default gw 192.168.0.253
```

And that's it! The tap device within the vps will get a random MAC address and will be visible to any other machines on the same network segment.

Subject: Re: Giving a VPS a MAC address
Posted by kir on Sat, 04 Mar 2006 21:44:37 GMT
View Forum Message <> Reply to Message

Excellent work! You might enhance its appearance by using CODE tags (in square brackets) for the pieces of code.

Is it possible to do multicasts/broadcasts inside a VPS set up that way? Have you tried to estimate the overhead?

Subject: Re: Giving a VPS a MAC address
Posted by Jason Stubbs on Sun, 05 Mar 2006 14:33:48 GMT
View Forum Message <> Reply to Message

Here's a few samples from tcpdump:

```
14:36:06.308917 (NOV-ETHII) IPX 00000000.00:00:74:64:91:ee.4100 >
00000000.ff:ff:ff:ff:ff:ff.0452: ipx-sap-nearest-req FileServer
14:36:06.883130 (NOV-ETHII) IPX 00000000.00:00:74:64:91:ee.4100 >
00000000.ff:ff:ff:ff:ff:ff.0452: ipx-sap-nearest-req FileServer
14:36:07.123994 802.1d config 8000.02:ff:ee:67:3d:bd.8001 root 8000.02:ff:ee:67:3d:bd pathcost
0 age 0 max 8 hello 2 fdelay 5
14:36:07.445022 (NOV-ETHII) IPX 00000000.00:00:74:64:91:ee.4100 >
00000000.ff:ff:ff:ff:ff:ff.0452: ipx-sap-nearest-req FileServer
```

14:36:09.123996 802.1d config 8000.02:ff:ee:67:3d:bd.8001 root 8000.02:ff:ee:67:3d:bd pathcost 0 age 0 max 8 hello 2 fdelay 5
14:36:11.123968 802.1d config 8000.02:ff:ee:67:3d:bd.8001 root 8000.02:ff:ee:67:3d:bd pathcost 0 age 0 max 8 hello 2 fdelay 5
14:36:11.878039 IP 192.168.1.50.netbios-dgm > 192.168.1.255.netbios-dgm: NBT UDP PACKET(138)
14:36:11.878270 IP 192.168.1.133.32768 > 192.168.1.1.domain: 41817+[|domain]
14:36:11.890160 IP 192.168.1.1.domain > 192.168.1.133.32768: 41817 NXDomain[|domain]
14:36:13.123912 802.1d config 8000.02:ff:ee:67:3d:bd.8001 root 8000.02:ff:ee:67:3d:bd pathcost 0 age 0 max 8 hello 2 fdelay 5
14:36:15.123915 802.1d config 8000.02:ff:ee:67:3d:bd.8001 root 8000.02:ff:ee:67:3d:bd pathcost 0 age 0 max 8 hello 2 fdelay 5
14:36:16.877573 arp who-has 192.168.1.1 tell 192.168.1.133
14:36:16.878623 arp reply 192.168.1.1 is-at 00:20:ed:20:35:ce (oui Unknown)
14:36:16.987026 arp who-has 192.168.1.50 tell 192.168.1.12

The OpenVZ test machine has the addresses 192.168.1.9 and .133 on a /24 network. The "NBT UDP PACKET" looks to be broadcast traffic. Seeing as there's IPX and ARP traffic getting through, there shouldn't be any problems otherwise either.

Those "802.1d config" are coming in once every two seconds. I'm not sure what that is about. I'll look into that as well as what overhead there is tomorrow.

Subject: Re: Giving a VPS a MAC address
Posted by Jason Stubbs on Mon, 06 Mar 2006 07:08:56 GMT
View Forum Message <> Reply to Message

Okay, now I'm surprised. I haven't tried to check change in CPU usage, but kernel memory usage does increase. The surprising thing though is that performance increases as well.

The figures below are from apache benchmark. The machine running apache benchmark is maxed out on the 75 byte file test as the test server is the fastest machine currently in the office. In both cases running through vtun is slightly faster.

Without VTun
kmemsize maxheld 2506316

75 byte file
Requests per second: 960.12 [#/sec] (mean)
Time per request: 5.208 [ms] (mean)
Time per request: 1.042 [ms] (mean, across all concurrent requests)
Transfer rate: 288.80 [Kbytes/sec] received

51,200 byte file
Requests per second: 166.94 [#/sec] (mean)
Time per request: 29.950 [ms] (mean)

Time per request: 5.990 [ms] (mean, across all concurrent requests)
Transfer rate: 8386.21 [Kbytes/sec] received

With VTun
kmemsize maxheld 2626204

75 byte file
Requests per second: 987.08 [#/sec] (mean)
Time per request: 5.065 [ms] (mean)
Time per request: 1.013 [ms] (mean, across all concurrent requests)
Transfer rate: 296.91 [Kbytes/sec] received

51,200 byte file
Requests per second: 168.39 [#/sec] (mean)
Time per request: 29.693 [ms] (mean)
Time per request: 5.939 [ms] (mean, across all concurrent requests)
Transfer rate: 8459.60 [Kbytes/sec] received

Is there any explanation for this? Lack of state table lookups or the like perhaps?

Subject: Re: Giving a VPS a MAC address
Posted by Jason Stubbs on Mon, 22 May 2006 06:38:20 GMT
View Forum Message <> Reply to Message

I'm now running this setup in production combined with LVS to give complete failover and load balancing without any idle hardware.

Now that there's a bit of traffic running over it though, overhead is a little more apparent. On a Xeon 2.4Ghz w/ HT (32bit/512kB cache), both the vtun client and server processes each use 1% CPU per 5MB (megabyte) / sec of bandwidth. That's taken when scp'ing a large file. Assuming packet size of 1460 bytes, that's about 4000 packets/sec. That gives about 200,000 packets/sec to exhaust a CPU. Likely a limit that won't be hit, but it does knock a small whole in the machine's computing power.

On the upside, I was afraid that with several tunnels being tied to the one bridge that all traffic would be sent across all tunnels making the above problem exponential. Luckily, Linux's bridging code seems to be smart enough to keep track of ARP replies and only send traffic where it needs to go.

Subject: Re: Giving a VPS a MAC address
Posted by dev on Fri, 26 May 2006 14:33:33 GMT
View Forum Message <> Reply to Message

Really soon we are releasing veth device with MAC. So I suppose it won't be an issue any more for you?

Subject: Re: Giving a VPS a MAC address
Posted by Jason Stubbs on Sat, 27 May 2006 04:57:15 GMT
View Forum Message <> Reply to Message

Yep, that would be wonderful. It would clear up all the issues I've been having.

Subject: Re: Giving a VPS a MAC address
Posted by dev on Wed, 28 Jun 2006 18:16:40 GMT
View Forum Message <> Reply to Message

In recent developement kernels this is also possible with veth network device.

See details in Wiki veth page:
http://wiki.openvz.org/Virtual_Ethernet_device