Kristian,

thanks for sharing this info.
However, since it looks like your problem is related to bonding and bridges
(not OpenVZ itself) I think you would be able to get quicker/better reply from
netdev@vger>@kernel.org mailing list. Please, keep this mail list on CC.

Thanks,
Kirill

>
>>Hi list,
>
> Hi list, will reply myelf :-)
>
>
>>I have a bonding/vlan/bridge/veth problem.
>>Sometimes a bridge think a veth device move to another port.
>>If I remove a physical interface from bond, the bridge behaves normally.
>>
>>Kernel 2.6.16 + openvz test020
>>VE0 Ubuntu dapper/6.06LTS, IP 172.31.1.26 on VLAN 254
>>VE1028 Debian stable/sarge/3.1, IP 10.1.28.12 on VLAN 28
>>
>>I have a server (vs5, VE0) using eth0 and eth1 in a bonding interface
>>bond0. bond0 is on tagged vlan.
>>I create a vlan device vlan254 on vlan 254. This is VE0 IP.
>>For each VE (XX) I do
>> create a vlan device vlanXX on vlan XX.
>> create a bridge bvXX and add vlanXX to it.
>> create a VE (VE10XX) using veth.
>> VETH="ve10XX.0,aa:00:04:56:YY:ZZ,eth0,aa:00:04:57:YY:ZZ"
>> add ve10XX.0 to the bridge.
>> YY and ZZ are calculated from VEID number (VLAN + 1000)
>>
>>   eth0   eth1
>>   \   /
>>   bond0
>>   / \   veth
>> vlan254   vlanXX   ve10XX.0 -- eth0 (ve10XX)
>> VE0   \   /
>>   bvXX (bridge)
>>

>
>
> The drawing above is correct, but the part not drawed
> is the important one.
>
> eth0 and eth1 are each connected to a switch.
> These are connected by trunk ports 1 and 2.
> The bond interface (eth0 + eth1) is in active/backup mode.
>
> When I ping 10.1.28.101 in vlan28 from ve1028 (10.1.28.12),
> it sends the following arp request:
> aa:00:04:57:04:04 > ff:ff:ff:ff:ff:ff arp who-has 10.1.28.101 tell 10.1.28.12
>
> The request will go from eth0 (VE1028) to ve1028.0 -> bv28 -> vlan28 ->
> bond0 -> eth0 -> SW1port16 -> SW1 ALL ports but 16 -> including SW2port1/2 ->
> SW2 ALL ports but 1/2 -> including target and eth1 -> bond0 -> vlan28 ->
> bv28 -> ve1028.0 -> eth0
>
> The target 10.28.1.101, receives the request through SW2 port 6.
> The switches/bridges gets updated as follows:
> bv28 know aa:00:04:57:04:04 is at port 2 (ve1028.0)
> SW1 know aa:00:04:57:04:04 is at port 16
> SW2 know aa:00:04:57:04:04 is at port 1/2
> bv28 know aa:00:04:57:04:04 is at port 1 (vlan28)
> Note bv28 gets updated twice.
>
> The target replies:
> 00:03:fa:0f:a3:a7 > aa:00:04:57:04:04 arp reply 10.1.28.101 is-at ...:0f:a3:a7
>
> The arp reply will go from SW2port6 -> SW2port1/2 -> SW1port1/2 ->
> SW1port16 -> eth0 -> bond0 -> vlan28 -> bv28 -> NULL
> As bv28 received the arp request from "aa:00:04:57:04:04" on port 1 (vlan28)
> it will not forward the arp reply to port 2 (ve1028.0), therefore eth0 in
> VE1028 never receives the arp reply... No communication.
>
> So the problem is bridging over bonding.
> The backup interface receives broadcast frames and forwards them to the bridge
> which updates its mac table.
>
> I will test the following.
>
>
>   SW1 ----- SW2
> | |
>   eth0   eth1
> | |
>   eth0.XX eth1.XX   vlan
>   \   /

```
>        bvXX          bridge
>          |
>        ve10XX.0          \
>          |          veth
>        eth0 (ve10XX)    /
>
> I just have to make sure to use spanning tree.
> The linux box should be in blocking mode.
>
> Comments?
>
> Regards,
> Kristian.
>
```