Subject: linux-2.6.20-openvz tree
Posted by Carl-Daniel Hailfinge on Thu, 22 Mar 2007 15:28:46 GMT

Hi,

I just noticed a new linux-2.6.20 git tree on git.openvz.org. Will
this tree eventually become a supported tree or is it just there as
preparation for an upstream merge?
Speaking of an upstream merge, is there any timeline for merging
network virtualization?

Regards,
Carl-Daniel

Subject: Re: linux-2.6.20-openvz tree
Posted by dev on Thu, 22 Mar 2007 15:45:48 GMT

our main stable branch right now is 2.6.18 based on RHEL5 kernel.

2.6.20 is a new development branch to be used for future releases and
feature developements. We also use it as a test ground for mainstream
namespaces support.

Speaking about upstream merges:
there are 2 network virtualization implementations currently exist.
I'm not sure how much time it will take to merge this work,
it is very much depends on netdev@ maintainers. Maybe 2-3 month.
Why are you interested in that? Do you want to use some particular feature?

Thanks,
Kirill

Carl-Daniel Hailfinger wrote:
> Hi,
>
> I just noticed a new linux-2.6.20 git tree on git.openvz.org. Will
> this tree eventually become a supported tree or is it just there as
> preparation for an upstream merge?
> Speaking of an upstream merge, is there any timeline for merging
> network virtualization?
>
> Regards,
> Carl-Daniel

Subject: Re: linux-2.6.20-openvz tree
Posted by Carl-Daniel Hailfinge on Thu, 22 Mar 2007 18:29:19 GMT
View Forum Message <> Reply to Message

On 22.03.2007 16:59, Kirill Korotaev wrote:
> Speaking about upstream merges:
> there are 2 network virtualization implementations currently exist.
> I'm not sure how much time it will take to merge this work,
> it is very much depends on netdev@ maintainers. Maybe 2-3 month.

OK, so the target is 2.6.22 or 2.6.23, if the usual time between
releases is used as a basis for the estimation.

> Why are you interested in that? Do you want to use some particular
> feature?

Yes. I currently use Linux policy routing for ONE machine performing
double/triple/... NAT. Many people state that this is impossible,
but it works fine unless two connections from the different subnets
have identical 5-tuples. In that case, the connection tracking code
gets confused. Unfortunately, the 5-tuple used by connection tracking
and NAT has no means to incorporate the NF mark, so I hope I can
use different containers for that.

However, last time I checked, all network virtualization attempts
did NOT consider one aspect I consider important for double NAT and
virtual routers: Efficiency. Once I use virtualization, I am
constrained to virtual network interfaces and suffer the overhead
of multiple routing/bridging decisions for one packet.
It would be great if I could make physical interfaces accessible
in a VE without resorting to bridging or routing. For example,
move eth0 and eth1 to one VE, eth2 and eth3 to another VE and
keep eth4 under control of the HN.
I admit that most of this can be done with policy routing and NF
marks, but connection tracking cares about neither of them.

Regards,
Carl-Daniel
--
http://www.hailfinger.org/

Subject: Re: linux-2.6.20-openvz tree
Posted by dev on Fri, 23 Mar 2007 08:51:47 GMT
View Forum Message <> Reply to Message

Carl-Daniel Hailfinger wrote:
> On 22.03.2007 16:59, Kirill Korotaev wrote:

>
>>Speaking about upstream merges:
>>there are 2 network virtualization implementations currently exist.
>>I'm not sure how much time it will take to merge this work,
>>it is very much depends on netdev@ maintainers. Maybe 2-3 month.
>
>
> OK, so the target is 2.6.22 or 2.6.23, if the usual time between
> releases is used as a basis for the estimation.
>
>
>>Why are you interested in that? Do you want to use some particular
>>feature?
>
>
> Yes. I currently use Linux policy routing for ONE machine performing
> double/triple/... NAT. Many people state that this is impossible,
> but it works fine unless two connections from the different subnets
> have identical 5-tuples. In that case, the connection tracking code
> gets confused. Unfortunately, the 5-tuple used by connection tracking
> and NAT has no means to incorporate the NF mark, so I hope I can
> use different containers for that.
>
> However, last time I checked, all network virtualization attempts
> did NOT consider one aspect I consider important for double NAT and
> virtual routers: Efficiency. Once I use virtualization, I am
> constrained to virtual network interfaces and suffer the overhead
> of multiple routing/bridging decisions for one packet.
> It would be great if I could make physical interfaces accessible
> in a VE without resorting to bridging or routing. For example,
> move eth0 and eth1 to one VE, eth2 and eth3 to another VE and
> keep eth4 under control of the HN.
This was possible for years in OpenVZ:

man vzctl
http://openvz.org/documentation/mans/vzctl.8

Network devices control parameters
--netdev_add name
    move network device from VE0 to a specified VE
--netdev_del name
    delete network device from a specified VE

this is exactly the thing you are talking about:
you can move eth0 and eth1 to one VE, eth2 and eth3 to another VE
and keep eth4 to HN.

And sure, this removes overhead of virtual network devices, additional

routing/bridging etc.
At the same time you can use separate NAT/firewall,routing,arp tables
inside each VE.

Isn't it the thing you want?

> I admit that most of this can be done with policy routing and NF
> marks, but connection tracking cares about neither of them.

Regards,
Kirill

Subject: Re: linux-2.6.20-openvz tree
Posted by Carl-Daniel Hailfinge on Sat, 24 Mar 2007 12:14:35 GMT
View Forum Message <> Reply to Message

On 23.03.2007 10:01, Kirill Korotaev wrote:
> This was possible for years in OpenVZ:
>
> man vzctl
> http://openvz.org/documentation/mans/vzctl.8
>
> Network devices control parameters
> --netdev_add name
>   move network device from VE0 to a specified VE
> --netdev_del name
>   delete network device from a specified VE
>
> this is exactly the thing you are talking about:
> you can move eth0 and eth1 to one VE, eth2 and eth3 to another VE
> and keep eth4 to HN.

Great! The OpenVZ wiki seemed to suggest that this was not possible.
Next time I'll read the man pages and not only the wiki before
asking questions.

Regards,
Carl-Daniel

Subject: Re: linux-2.6.20-openvz tree
Posted by kir on Sat, 24 Mar 2007 16:01:02 GMT
View Forum Message <> Reply to Message

Carl-Daniel Hailfinger wrote:
> On 23.03.2007 10:01, Kirill Korotaev wrote:

>
>> This was possible for years in OpenVZ:
>>
>> man vzctl
>> http://openvz.org/documentation/mans/vzctl.8
>>
>> Network devices control parameters
>> --netdev_add name
>>   move network device from VE0 to a specified VE
>> --netdev_del name
>>   delete network device from a specified VE
>>
>> this is exactly the thing you are talking about:
>> you can move eth0 and eth1 to one VE, eth2 and eth3 to another VE
>> and keep eth4 to HN.
>>
>
> Great! The OpenVZ wiki seemed to suggest that this was not possible.
>

Can you fix this wiki page? Or at least point me to it, so I can fix?

> Next time I'll read the man pages and not only the wiki before
> asking questions.
>

Subject: Re: linux-2.6.20-openvz tree
Posted by Carl-Daniel Hailfinge on Tue, 27 Mar 2007 17:40:16 GMT
View Forum Message <> Reply to Message

On 24.03.2007 17:01, Kir Kolyshkin wrote:
> Carl-Daniel Hailfinger wrote:
>> On 23.03.2007 10:01, Kirill Korotaev wrote:
>>
>>> This was possible for years in OpenVZ:
>>>
>>> man vzctl
>>> http://openvz.org/documentation/mans/vzctl.8
>>>
>>> Network devices control parameters
>>> --netdev_add name
>>>   move network device from VE0 to a specified VE --netdev_del name
>>>   delete network device from a specified VE
>>>
>>> this is exactly the thing you are talking about:
>>> you can move eth0 and eth1 to one VE, eth2 and eth3 to another VE
>>> and keep eth4 to HN.

>>>
>>
>> Great! The OpenVZ wiki seemed to suggest that this was not possible.
>>
>
> Can you fix this wiki page? Or at least point me to it, so I can fix?

http://wiki.openvz.org/Differences_between_venet_and_veth states
"OpenVZ provides you to use either venet or veth devices (or both) for
in-VE networking."
Suggested new wording:
"OpenVZ provides you to use venet or veth or real ethernet (ethX)
devices (or a combination thereof) for in-VE networking."

This article and the others in Category:Networking never mention the
ability to move ethX devices to a VE.

Regards,
Carl-Daniel