Subject: openvz ploop images on moosefs mount
Posted by Corin Langosch on Sat, 31 Mar 2012 18:44:56 GMT
View Forum Message <> Reply to Message

Hi,

has anybody tried using the new ploop storage for openvz images together
with moosefs (http://www.moosefs.org/)?

ploop mount -d /dev/ploop0 /mfs-mount/root.hdd
Adding delta dev=/dev/ploop0 img=/mfs-mount/root.hdd (rw)
PLOOP_IOC_ADD_DELTA /mfs-mount/root.hdd: Invalid argument

In syslog I find:
kernel: File on FS without backing device

I thought it might be because ploop needs direct-io (right?), but
mounting with direct-io enabled seems not to be supported by moosefs:

mfsmount -o direct_io /mfs-mount
mfsmaster accepted connection with parameters: read-write,restricted_ip
; root mapped to root:root
fuse: unknown option `direct_io'

Has anybody got it working somehow? :)

Btw: I'll also post this question to the moosefs mailinglist as I'm
pretty sure the devs there might have interesting answers too.

Corin

Subject: Re: openvz ploop images on moosefs mount
Posted by Kirill Kolyshkin on Sat, 31 Mar 2012 18:54:17 GMT
View Forum Message <> Reply to Message

Ploop only supports ext4 and nfs

<info@corinlangosch.com>

> Hi,
>
> has anybody tried using the new ploop storage for openvz images together
> with moosefs (http://www.moosefs.org/)?
>
> ploop mount -d /dev/ploop0 /mfs-mount/root.hdd
> Adding delta dev=/dev/ploop0 img=/mfs-mount/root.hdd (rw)

> PLOOP_IOC_ADD_DELTA /mfs-mount/root.hdd: Invalid argument
>
> In syslog I find:
> kernel: File on FS without backing device
>
> I thought it might be because ploop needs direct-io (right?), but mounting
> with direct-io enabled seems not to be supported by moosefs:
>
> mfsmount -o direct_io /mfs-mount
> mfsmaster accepted connection with parameters: read-write,restricted_ip ;
> root mapped to root:root
> fuse: unknown option `direct_io'
>
> Has anybody got it working somehow? :)
>
> Btw: I'll also post this question to the moosefs mailinglist as I'm pretty
> sure the devs there might have interesting answers too.
>
> Corin
>
>

Subject: Re: openvz ploop images on moosefs mount
Posted by Corin Langosch on Sat, 31 Mar 2012 19:13:25 GMT
View Forum Message <> Reply to Message

Hi Kirill,

thank you for your very fast response. :)

Am 31.03.2012 20:54, schrieb Kirill Kolyshkin:
>
> Ploop only supports ext4 and nfs
>
Is there any special reason for this? I'd really like to know the
technical aspects because I then might be able to make moosefs (or
another fuse based fs developed by myself) working with ploop. If I can
read about it in any documents, please redirect me to them :)

Another question: is it (easily) possible make an openvz container use a
normal block devices for storage (which is formatted with ex. ext4)?
Before seeing ploop I thought this won't be possible, but now with ploop
it seems vzctl is only mounting /dev/ploopX somewhere and then using
this mount as the container's root. I'd especially be interested in
using ex. an iscsi backed block device with openvz containers.

Corin

Am 31.03.2012 20:54, schrieb Kirill Kolyshkin:
>
> Ploop only supports ext4 and nfs
>
> 31.03.2012 22:50 ????????????? "Corin Langosch" <info@corinlangosch.com
> <mailto:info@corinlangosch.com>> ???????:
>
> Hi,
>
> has anybody tried using the new ploop storage for openvz images
> together with moosefs (http://www.moosefs.org/)?
>
> ploop mount -d /dev/ploop0 /mfs-mount/root.hdd
> Adding delta dev=/dev/ploop0 img=/mfs-mount/root.hdd (rw)
> PLOOP_IOC_ADD_DELTA /mfs-mount/root.hdd: Invalid argument
>
> In syslog I find:
> kernel: File on FS without backing device
>
> I thought it might be because ploop needs direct-io (right?), but
> mounting with direct-io enabled seems not to be supported by moosefs:
>
> mfsmount -o direct_io /mfs-mount
> mfsmaster accepted connection with parameters:
> read-write,restricted_ip ; root mapped to root:root
> fuse: unknown option `direct_io'
>
> Has anybody got it working somehow? :)
>
> Btw: I'll also post this question to the moosefs mailinglist as
> I'm pretty sure the devs there might have interesting answers too.
>
> Corin
>
>
> _____
> Users mailing list
> Users@openvz.org <mailto:Users@openvz.org>
> https://openvz.org/mailman/listinfo/users
>
>
>

Subject: Re: openvz ploop images on moosefs mount
Posted by kir on Sun, 01 Apr 2012 12:11:12 GMT

On 03/31/2012 11:13 PM, Corin Langosch wrote:
> Hi Kirill,
>
> thank you for your very fast response. :)
>
> Am 31.03.2012 20:54, schrieb Kirill Kolyshkin:
>>
>> Ploop only supports ext4 and nfs
>>
> Is there any special reason for this? I'd really like to know the
> technical aspects because I then might be able to make moosefs (or
> another fuse based fs developed by myself) working with ploop. If I
> can read about it in any documents, please redirect me to them :)

Ploop developers are now working on fuse I/O module for ploop (and fuse improvements required for it). When this work will be ready, you might revisit this issue and see what is required from moosefs to be used for ploop.

>
> Another question: is it (easily) possible make an openvz container use
> a normal block devices for storage (which is formatted with ex. ext4)?
> Before seeing ploop I thought this won't be possible, but now with
> ploop it seems vzctl is only mounting /dev/ploopX somewhere and then
> using this mount as the container's root. I'd especially be interested
> in using ex. an iscsi backed block device with openvz containers.

Yes it's pretty possible, with some minor modifications to vzctl. I need to think a bit more about it, but it looks like a trivial case (no vzquota etc). Let me know if you want to work on that.

>
> Am 31.03.2012 20:54, schrieb Kirill Kolyshkin:
>>
>> Ploop only supports ext4 and nfs
>>

>>
>> Hi,
>>
>> has anybody tried using the new ploop storage for openvz images
>> together with moosefs (http://www.moosefs.org/)?
>>
>> ploop mount -d /dev/ploop0 /mfs-mount/root.hdd
>> Adding delta dev=/dev/ploop0 img=/mfs-mount/root.hdd (rw)
>> PLOOP_IOC_ADD_DELTA /mfs-mount/root.hdd: Invalid argument

>>
>> In syslog I find:
>> kernel: File on FS without backing device
>>
>> I thought it might be because ploop needs direct-io (right?), but
>> mounting with direct-io enabled seems not to be supported by moosefs:
>>
>> mfsmount -o direct_io /mfs-mount
>> mfsmaster accepted connection with parameters:
>> read-write,restricted_ip ; root mapped to root:root
>> fuse: unknown option `direct_io'
>>
>> Has anybody got it working somehow? :)
>>
>> Btw: I'll also post this question to the moosefs mailinglist as
>> I'm pretty sure the devs there might have interesting answers too.
>>
>> Corin
>>

Subject: Re: openvz ploop images on moosefs mount
Posted by Aleksandar Ivanisevic on Mon, 02 Apr 2012 10:19:02 GMT
View Forum Message <> Reply to Message

Corin Langosch
<info@corinlangosch.com> writes:

> Hi,
>
> has anybody tried using the new ploop storage for openvz images
> together with moosefs (http://www.moosefs.org/)?

Out of interest, have you tried running VEs off of moosefs, directly,
without ploop? How does that work, if at all?

I'm still trying to find a replacement for DRBD for a HA setup with a
lower latency, but all I find when I google "moosefs openvz" is my own
question on this mailing list a year ago ;)

Subject: Re: Re: openvz ploop images on moosefs mount
Posted by Corin Langosch on Mon, 02 Apr 2012 10:49:49 GMT
View Forum Message <> Reply to Message

Am 02.04.2012 12:19, schrieb Aleksandar Ivanisevic:
> Out of interest, have you tried running VEs off of moosefs, directly,

> without ploop? How does that work, if at all?

Yes, but only for a quick test and not with any production data:

- setup moosefs as usual and mount it on host
- create (sparse) file for container on moosefs mount
- use this file to setup a loop device using losetup
- make fs on loop device and mount it to the container's private folder
- start the container as usual

I din't really like this approach because it involes a lot of (slow) layers and double buffering issues and so I'd expect quite bad performance. If you try it yourself and do some benchmarks I'd be very happy to know the results.

I didn't try to use moosefs directly (without loop device) as a backing storage for containers because I made very bad experience (mostly performance and stability) with all kinds of network storages in the past.

From my past experience having a distributed block device (and not a distributed file system) is the way to go, as the system then has almost "native" performance due to the page cache (if not too many layers are involved as above).

I'm currently evaluating distributed network block devices, but unluckily there doesn't one to exist yet:
http://serverfault.com/questions/375570/distributed-fault-tolerant-network-block-device/375821#375821

Corin

Subject: Re: Re: openvz ploop images on moosefs mount
Posted by Corin Langosch on Mon, 02 Apr 2012 10:54:35 GMT
View Forum Message <> Reply to Message

Am 02.04.2012 12:19, schrieb Aleksandar Ivanisevic:
> Out of interest, have you tried running VEs off of moosefs, directly,
> without ploop? How does that work, if at all?

Yes, but only for a quick test and not with any production data:

- setup moosefs as usual and mount it on host
- create (sparse) file for container on moosefs mount
- use this file to setup a loop device using losetup
- make fs on loop device and mount it to the container's private folder
- start the container as usual

I din't really like this approach because it involes a lot of (slow) layers and double buffering issues and so I'd expect quite bad performance. If you try it yourself and do some benchmarks I'd be very happy to know the results.

I didn't try to use moosefs directly (without loop device) as a backing storage for containers because I made very bad experience (mostly performance and stability) with all kinds of network storages in the past.

From my past experience having a distributed block device (and not a distributed file system) is the way to go, as the system then has almost "native" performance due to the page cache (if not too many layers are involved as above).

I'm currently evaluating distributed network block devices, but unluckily there doesn't one to exist yet:
http://serverfault.com/questions/375570/distributed-fault-tolerant-network-block-device/375821#375821

Corin

Subject: Re: openvz ploop images on moosefs mount
Posted by Corin Langosch on Mon, 02 Apr 2012 11:56:26 GMT
View Forum Message <> Reply to Message

Hi Kir,

Am 01.04.2012 14:11, schrieb Kir Kolyshkin:
>
> Ploop developers are now working on fuse I/O module for ploop (and
> fuse improvements required for it). When this work will be ready, you
> might revisit this issue and see what is required from moosefs to be
> used for ploop.

Sounds great. Can you please point me to the current git repo of the kernel stuff? The kernel stuff at http://git.openvz.org/ seems to be outdated and I couldn't find the new address.

>
> Yes it's pretty possible, with some minor modifications to vzctl. I
> need to think a bit more about it, but it looks like a trivial case
> (no vzquota etc). Let me know if you want to work on that.

Yes I could try, but I cannot guarentee how much time I can invest. Please let me know how to proceed if anything special is needed. Otherwise I'd just clone the git repo and send you the patch when finished. Or I'd clone the repo at github and create a new branch for

the feature?

Corin

Subject: Re: openvz ploop images on moosefs mount
Posted by kir on Mon, 02 Apr 2012 18:23:32 GMT
View Forum Message <> Reply to Message

On 04/02/2012 03:56 PM, Corin Langosch wrote:
> Hi Kir,
>
> Am 01.04.2012 14:11, schrieb Kir Kolyshkin:
>>
>> Ploop developers are now working on fuse I/O module for ploop (and
>> fuse improvements required for it). When this work will be ready, you
>> might revisit this issue and see what is required from moosefs to be
>> used for ploop.
>
> Sounds great. Can you please point me to the current git repo of the
> kernel stuff? The kernel stuff at http://git.openvz.org/ seems to be
> outdated and I couldn't find the new address.

Since we haven't yet tried to push ploop upstream yet its git repo is
not available (and internally we use something like stgit but based on
CVS and lots of shell code and *.patch files).

>
>>
>> Yes it's pretty possible, with some minor modifications to vzctl. I
>> need to think a bit more about it, but it looks like a trivial case
>> (no vzquota etc). Let me know if you want to work on that.
>
> Yes I could try, but I cannot guarentee how much time I can invest.
> Please let me know how to proceed if anything special is needed.
> Otherwise I'd just clone the git repo and send you the patch when
> finished. Or I'd clone the repo at github and create a new branch for
> the feature?

A patch (or, better, a patchset) is fine.

I suggest you start with using --layout option, introduce a new CT
layout (say 'device'... can't think of any good name right now) and
patch the mount/umount code to support it. VE_PRIVATE can be a block
device name in that case I guess, VE_ROOT remains the same (ie mount point).
