Subject: How to use binfmt_misc in VE
Posted by ramjet on Tue, 13 Oct 2009 13:14:03 GMT
View Forum Message <> Reply to Message

I would like to register a binary handler for byte compiled python code.

lenny2:/# mount -t binfmt_misc none /proc/sys/fs/binfmt_misc
mount: unknown filesystem type 'binfmt_misc'

lenny2:/# mount
simfs on / type simfs (rw)
tmpfs on /lib/init/rw type tmpfs (rw,nosuid,mode=755)
proc on /proc type proc (rw,nosuid,nodev,noexec)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
devpts on /dev/pts type devpts (rw,nosuid,noexec,gid=5,mode=620)

On hardware node I'm running Debian GNU/Linux 5.0.3 with kernel 2.6.26-2-openvz-amd64 and vzctl 3.0.23-8

On a proxmox PVE 1.1 with kernel 2.6.24-2-pve with vzctl 3.0.23-1pve1 I get same error.

I have tried loading binfmt_misc and mounting binfmt_misc on HN and restarted VE without luck.

When googling for OpenVZ and binfmt_misc I can see that other people have fs type binfmt_misc mounted inside VE.

Is binfmt_misc handler interface supported inside VE's?
Is it just a capability setting?
Am I running kernels which are missing a patch?

I need a recent kernel on HN because of RAID driver

Subject: Re: How to use binfmt_misc in VE
Posted by ramjet on Thu, 15 Oct 2009 16:03:20 GMT
View Forum Message <> Reply to Message

I compiled a 2.6.27 kernel (http://wiki.openvz.org/Download/kernel/2.6.27/2.6.27-briullo v.1) using vanilla Linux 2.6.27 kernel and patch-briullov.1-combined.gz.

But I'm still not able to get binfmt_misc listed in the VE in /proc/filesystems.

From this post I can see that it's possible:
 http://forum.openvz.org/index.php?t=tree&&th=7839&am p;goto=37166#msg_37166

So what's the trick? Please...

Subject: Re: How to use binfmt_misc in VE
Posted by maratrus on Sat, 24 Oct 2009 13:17:54 GMT
View Forum Message <> Reply to Message

Dear Peter,

I'm terribly sorry for that big delay.
There is no trick at all.
binfmt_misc can be mounted with RHEL5 based kernels
(look at fs/binfmt_misc.c - the part under #ifdef CONFIG_VE).
Please use that kernels if you have a chance.
Unfortunately, 2.6.26 kernels don't have such a virtualization trick so I would recommend you to
file a bug (mark it as a feature request for example).
Thank you.