Subject: Q: hardcoded parameters and restrictions
Posted by ldv on Tue, 15 Aug 2006 23:45:34 GMT
View Forum Message <> Reply to Message

Hi,

vzctl contains a few hardcoded parameters and restrictions which need to be converted into configurable vps.conf parameters.
In particular, I mean HOME and PATH environment variables, allowed fs types (e.g. sysfs) - all these parameters should be easy to implement.

kernel space also contains hardcoded stuff.
There is a flag to enable/disable sysfs within container, but I found no normal way to disable /proc.
Also, it seems to be no way to disable devices listed in default_minor_perms.
Set of files added to virtualized /proc is also hardcoded, I found no way to e.g. add /proc/devices file required for some third party software.

--
ldv

Subject: Re: Q: hardcoded parameters and restrictions
Posted by Kirill Korotaev on Tue, 22 Aug 2006 11:04:20 GMT
View Forum Message <> Reply to Message

Hello Dmitry,

> Hi,
>
> vzctl contains a few hardcoded parameters and restrictions which need to be
> converted into configurable vps.conf parameters.
> In particular, I mean HOME and PATH environment variables,
do you mean HOME and PATH which are provided to VPS init?
mmm, probably can be made configurable from vps.conf
do you think it is required?

> allowed fs types (e.g. sysfs) -
sysfs was disabled for 2.6.8 kernels (by default)
only due to unreclaimable sysfs memory which took
about ~0.5-1Mb of RAM per VE.

It will be enabled for newer kernels by default.

> all these parameters should be easy to implement.

> kernel space also contains hardcoded stuff.

> There is a flag to enable/disable sysfs within container, but I found no
> normal way to disable /proc.
because sysfs was done disabled by default for memory consumption reason :/
/proc has no this problem and is always enabled.

> Also, it seems to be no way to disable devices listed in default_minor_perms.
applications do not work w/o /dev/null and others at all :)

> Set of files added to virtualized /proc is also hardcoded, I found no way
> to e.g. add /proc/devices file required for some third party software.
this one is really the most usefull imho.

in general you are right, many of these can be generalized and made
more tunable. Will appreciate any help on this and will do my best
to help and discuss what and how is needed.

Thanks,
Kirill

Subject: Re: Q: hardcoded parameters and restrictions
Posted by ldv on Tue, 22 Aug 2006 11:56:05 GMT
View Forum Message <> Reply to Message

Hi,

On Tue, Aug 22, 2006 at 03:06:56PM +0400, Kirill Korotaev wrote:
> Hello Dmitry,
>
> >vzctl contains a few hardcoded parameters and restrictions which need to be
> >converted into configurable vps.conf parameters.
> >In particular, I mean HOME and PATH environment variables,
> do you mean HOME and PATH which are provided to VPS init?

Yes, I mean VPS init, vzctl exec and vzctl enter.

> mmm, probably can be made configurable from vps.conf
> do you think it is required?

I think this is required since each distro has own PATH policy.

> >allowed fs types (e.g. sysfs) -
> sysfs was disabled for 2.6.8 kernels (by default)
> only due to unreclaimable sysfs memory which took
> about ~0.5-1Mb of RAM per VE.
>
> It will be enabled for newer kernels by default.
>

> >all these parameters should be easy to implement.
>
> >kernel space also contains hardcoded stuff.
> >There is a flag to enable/disable sysfs within container, but I found no
> >normal way to disable /proc.
> because sysfs was done disabled by default for memory consumption reason :/
> /proc has no this problem and is always enabled.

Unfortunately, /proc is notorious to had security-related bugs in the past,
including arbitrary code execution in kernel space. Since I'm not sure
that all such bugs are fixed, and since not all tasks require /proc to be
mounted, I'd like to be able to disable /procfs on per-VPS basis.

> >Also, it seems to be no way to disable devices listed in
> >default_minor_perms.
> applications do not work w/o /dev/null and others at all :)

Yes, /dev/null and /dev/zero are not an issue.
I care about /dev/random; how to deal with potential lack of randomness
in the system?

> >Set of files added to virtualized /proc is also hardcoded, I found no way
> >to e.g. add /proc/devices file required for some third party software.
> this one is really the most usefull imho.
>
> in general you are right, many of these can be generalized and made
> more tunable. Will appreciate any help on this and will do my best
> to help and discuss what and how is needed.

I think I could help with patches to userspace code if necessary.

--
ldv

Subject: Re: Q: hardcoded parameters and restrictions
Posted by dev on Thu, 24 Aug 2006 11:27:53 GMT
View Forum Message <> Reply to Message

Dmitry V. Levin wrote:
> Hi,
>
> On Tue, Aug 22, 2006 at 03:06:56PM +0400, Kirill Korotaev wrote:
>
>>Hello Dmitry,
>>
>>

>>>vzctl contains a few hardcoded parameters and restrictions which need to be
>>>converted into configurable vps.conf parameters.
>>>In particular, I mean HOME and PATH environment variables,
>>
>>do you mean HOME and PATH which are provided to VPS init?
>
>
> Yes, I mean VPS init, vzctl exec and vzctl enter.
>
>
>>mmm, probably can be made configurable from vps.conf
>>do you think it is required?
>
>
> I think this is required since each distro has own PATH policy.
from kernel init/main.c:
static char * argv_init[MAX_INIT_ARGS+2] = { "init", NULL, };
char * envp_init[MAX_INIT_ENVS+2] = { "HOME=/", "TERM=linux", NULL, };

i.e. each init is run almost w/o any environment.
the same for VE.

on VE enter bash initializes PATH, HOME according to it's scripts.
So the only "bad" case I see is vzctl exec, right?

>>>allowed fs types (e.g. sysfs) -
>>
>>sysfs was disabled for 2.6.8 kernels (by default)
>>only due to unreclaimable sysfs memory which took
>>about ~0.5-1Mb of RAM per VE.
>>
>>It will be enabled for newer kernels by default.
>>
>>
>>>all these parameters should be easy to implement.
>>
>>>kernel space also contains hardcoded stuff.
>>>There is a flag to enable/disable sysfs within container, but I found no
>>>normal way to disable /proc.
>>
>>because sysfs was done disabled by default for memory consumption reason :/
>>/proc has no this problem and is always enabled.
>
>
> Unfortunately, /proc is notorious to had security-related bugs in the past,
> including arbitrary code execution in kernel space. Since I'm not sure
> that all such bugs are fixed, and since not all tasks require /proc to be
> mounted, I'd like to be able to disable /procfs on per-VPS basis.

Modern glibc (at least version from FC5) even doesn't work w/o /proc.
We had a bug with cp due to this :/
But in general I don't mind to make everything configurable.

How do you see it? via the same "features" mask as done with sysfs?
any other features? I think it is better to create a set of
env_create funtions allowing vzctl to control which features a initialized in VE.

>>>Also, it seems to be no way to disable devices listed in
>>>default_minor_perms.
>>
>>applications do not work w/o /dev/null and others at all :)
>
>
> Yes, /dev/null and /dev/zero are not an issue.
> I care about /dev/random; how to deal with potential lack of randomness
> in the system?
Good point.
let's move this into vzctl?
lets start from this one.
http://bugzilla.openvz.org/show_bug.cgi?id=241

>>>Set of files added to virtualized /proc is also hardcoded, I found no way
>>>to e.g. add /proc/devices file required for some third party software.
>>
>>this one is really the most usefull imho.
>>
>>in general you are right, many of these can be generalized and made
>>more tunable. Will appreciate any help on this and will do my best
>>to help and discuss what and how is needed.
>
>
> I think I could help with patches to userspace code if necessary.
thanks a lot!

Thanks,
Kirill

Subject: Re: Q: hardcoded parameters and restrictions
Posted by ldv on Thu, 24 Aug 2006 20:56:23 GMT
View Forum Message <> Reply to Message

Hi,

On Thu, Aug 24, 2006 at 03:30:38PM +0400, Kirill Korotaev wrote:

> Dmitry V. Levin wrote:
> >On Tue, Aug 22, 2006 at 03:06:56PM +0400, Kirill Korotaev wrote:
[...]
> >>>In particular, I mean HOME and PATH environment variables,
> >>
> >>do you mean HOME and PATH which are provided to VPS init?
> >
> >Yes, I mean VPS init, vzctl exec and vzctl enter.
> >
> >>mmm, probably can be made configurable from vps.conf
> >>do you think it is required?
> >
> >I think this is required since each distro has own PATH policy.
> from kernel init/main.c:
> static char * argv_init[MAX_INIT_ARGS+2] = { "init", NULL, };
> char * envp_init[MAX_INIT_ENVS+2] = { "HOME=/", "TERM=linux", NULL, };
>
> i.e. each init is run almost w/o any environment.
> the same for VE.

I agree, lets leave "init" case unchanged.

> on VE enter bash initializes PATH, HOME according to it's scripts.
> So the only "bad" case I see is vzctl exec, right?

Not only "exec", but also "enter", because bash does its initialization
depending on $HOME.

[...]
> >Unfortunately, /proc is notorious to had security-related bugs in the past,
> >including arbitrary code execution in kernel space. Since I'm not sure
> >that all such bugs are fixed, and since not all tasks require /proc to be
> >mounted, I'd like to be able to disable /procfs on per-VPS basis.
> Modern glibc (at least version from FC5) even doesn't work w/o /proc.
> We had a bug with cp due to this :/

No, that was a bug in FC5 coreutils package, fixed in their
coreutils-5.96-1.1 update.

> But in general I don't mind to make everything configurable.
>
> How do you see it? via the same "features" mask as done with sysfs?

To make just procfs configurable like sysfs, features mask should be enough.

> any other features? I think it is better to create a set of
> env_create funtions allowing vzctl to control which features a initialized
> in VE.

It depends on how much features will be made configurable.

> >>>Also, it seems to be no way to disable devices listed in
> >>>default_minor_perms.
> >>
> >>applications do not work w/o /dev/null and others at all :)
> >
> >Yes, /dev/null and /dev/zero are not an issue.
> >I care about /dev/random; how to deal with potential lack of randomness
> >in the system?
> Good point.
> let's move this into vzctl?
> lets start from this one.
> http://bugzilla.openvz.org/show_bug.cgi?id=241

OK

--
ldv
