Subject: UUID inside VE
Posted by mwojciechowski on Thu, 19 Aug 2010 07:16:18 GMT
View Forum Message <> Reply to Message

Hi,

OpenVZ VE root block device is /dev/simfs. Unfortunately I need to provide UUID of it for licensing purposes.

UUID is "connected" with udev which is not used inside OpenVZ VEs so this may be a problem. There is no /etc/blkid.tab either. I tried to create one manualy just for test if blkid will read from it but it doesn't.

Is there any solution for this other than use KVM? Maybe some other location to manually enter the "fake" UUID?

Thanx,
Martin.

Subject: Re: UUID inside VE
Posted by maratrus on Fri, 20 Aug 2010 08:38:51 GMT
View Forum Message <> Reply to Message

Hello,

Quote:
OpenVZ VE root block device is /dev/simfs. Unfortunately I need to provide UUID of it for licensing purposes.

sorry but I missed how exactly UUID is being checked?
Could you please explain in detail what kind of functionality
must be supported inside a VE?
Is a system_call being called? Or a file inside /dev is being checked? Or an output of some file must obey a particular rule?

Subject: Re: UUID inside VE
Posted by mwojciechowski on Fri, 20 Aug 2010 11:20:59 GMT
View Forum Message <> Reply to Message

maratrus wrote on Fri, 20 August 2010 10:38Hello,

Is a system_call being called? Or a file inside /dev is being checked? Or an output of some file must obey a particular rule?

License generator is using system_call from within libuuid.so and libblkid.so

Details about the server:

I deployed one OpenVZ VE based on Centos 5.2-amd64 template and passed access to it to the company that installed the PACS system (for storing diagnostic image data). Few days ago I received a message from them, that their licensing mechanism is based on UUID of the root block device.

Kernel: 2.6.18-2-pve
HN: Proxmox 1.5 (debian based)

Subject: Re: UUID inside VE
Posted by maratrus on Fri, 20 Aug 2010 12:36:03 GMT
View Forum Message <> Reply to Message

Is it possible to find out which system_call is called?
If there is an utility which fails the strace can be used.

But generally speaking, if some kernel functionality is used then
it's impossible to workaround without modification of the kernel.
