
Subject: Re: Hooks at start/stop of VE

Posted by [oscarah](#) on Tue, 19 Oct 2010 16:54:27 GMT

[View Forum Message](#) <> [Reply to Message](#)

Well, searching a bit more, I've found something. If you put an script called CTID.mount, on /etc/vz/conf, it will be called at mounting the specified CTID. There are a CTID.umount too.

The problem with this is that .umount is called AFTER umounting root filesystem. I want a script called on post-start and pre-stop of each container.

So, to solve this, I've written a small script which wraps the tool vzctl, and launches the hooks. The script is the following:

```
#!/bin/bash
# -*- mode: sh; coding: utf-8 -*-

# Copyright (c) 2010 Oscar Aceña.
# This is Free Software, under terms of GPL v3.0 (or later)

# NOTE: this will not check the correctness of parameters, so
# call it only correctly

# Execution synopsis
# $ vzctl [--help | --version]
# $ vzctl [--quiet | --verbose] <start | stop | restart> CTID

if [ -z "$VZCTL" ]; then VZCTL=/usr/sbin/vzctl-bin; fi
if [ -z "$HOOKSDIR" ]; then HOOKSDIR=/etc/vz/hooks; fi

if [ ! -d "$HOOKSDIR" ]; then
    $VZCTL "$@"
    echo "No hooks defined"
    exit $?
fi

if [ $# -gt 1 ] && [ "$1" != "--help" ] && [ "$1" != "--version" ]; then

    # --skiplock is not documented, but is used on /etc/init.d/vz
    if [ "$1" == "--verbose" ] || [ "$1" == "--quiet" ] || [ "$1" == "--skiplock" ]; then
        CMD="$2"
        CTID="$3"
    else
        CMD="$1"
        CTID="$2"
    fi

    if ! [ "$CTID" -eq "$CTID" 2> /dev/null ]; then
```

```

nCTID=$(vzlist "$CTID" 2> /dev/null | grep "$CTID" | awk '{print $1}')
if [ -z $nCTID ]; then
    echo "CT ID $CTID is invalid."
    exit 1
fi
else
nCTID=$CTID
fi

if vzlist $nCTID | grep -qs 'running'; then

if [ $CMD == "stop" ] || [ $CMD == "restart" ]; then
    hook="$HOOKSDIR/$nCTID.pre.stop"
    if [ -f "$hook" ]; then $hook; fi
fi

$VZCTL "$@"

if [ $CMD == "restart" ]; then
    hook="$HOOKSDIR/$nCTID.post.start"
    if [ -f "$hook" ]; then $hook; fi
fi
exit 0

else

$VZCTL "$@"

if [ $CMD == "start" ] || [ $CMD == "restart" ]; then
    hook="$HOOKSDIR/$nCTID.post.start"
    if [ -f "$hook" ]; then $hook; fi
fi
exit 0
fi

# In order to identify the wrapper
if [ -z "$@" ]; then echo -n "w-"; fi

$VZCTL "$@"
exit $?

```

It will call hooks on /etc/vz/hooks directory. The names of each hook is CTID.pre.stop and CTID.post.start. This files must have execution permission.

I'm using this script as an alternative (/etc/alternatives) for vzctl. You can install it using this makefile:

```
# -*- mode: make-gnumakefile; coding: utf-8 -*-
```

all:

install:

```
install -d $(DESTDIR)/usr/local/bin
install -d $(DESTDIR)/etc/alternatives
install -d $(DESTDIR)/usr/sbin
```

```
install -m 755 vzctl-wrapp.sh $(DESTDIR)/usr/local/bin
mv $(DESTDIR)/usr/sbin/vzctl $(DESTDIR)/usr/sbin/vzctl-bin
ln -fs $(DESTDIR)/usr/local/bin/vzctl-wrapp.sh $(DESTDIR)/etc/alternatives/vzctl
ln -fs $(DESTDIR)/etc/alternatives/vzctl $(DESTDIR)/usr/sbin
```

remove:

```
rm $(DESTDIR)/etc/alternatives/vzctl
rm $(DESTDIR)/usr/local/bin/vzctl-wrapp.sh
mv $(DESTDIR)/usr/sbin/vzctl-bin $(DESTDIR)/usr/sbin/vzctl
```

The makefile is for standard Debian systems. Change locations of files to match your system. To install, just "make install" as root. To remove, "make remove".

Hope it help you.

Cheers,

File Attachments

- 1) [vzctl-wrapp.sh](#), downloaded 266 times
 - 2) [Makefile](#), downloaded 303 times
-