
Subject: [WISH] exec early script at start

Posted by [Yoann Moulin](#) on Thu, 13 Dec 2007 11:13:11 GMT

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

Hi all,

I'm working on a xen to openvz migration, and I still have some issue to fix. but some seem to be not possible yet.

actually, on our diskless servers (xen and others), we are using nfsroot. And at start, in the initrd, we are using a script that make a connexion on the nfs boot server and remove the old root, create a new one, take the last system image build, and re apply configuration with cfengine tools.

I'd like to do the same with openvz, but I didn't see in the source this possibility.

it could be someting like in src/lib/env.c in the vps_start_custom function (in pseudo code) :

```
    memset(&actions, 0, sizeof(actions));
    if (check_var(res->fs.root, "VE_ROOT is not set"))
        return VZ_VE_ROOT_NOTSET;
    if (vps_is_run(h, veid)) {
        logger(-1, 0, "VE is already running");
        return VZ_VE_RUNNING;
    }
+
+   if (EARLY_SCRIPT) {
+       exec(EARLY_SCRIPT);
+   }
+
    if ((ret = check_ub(&res->ub)))
        return ret;
    dist_name = get_dist_name(&res->tpl);
    ret = read_dist_actions(dist_name, DIST_DIR, &actions);
    if (dist_name != NULL)
        free(dist_name);
    if (ret)
        return ret;
    logger(0, 0, "Starting VE ...");
    if (vps_is_mounted(res->fs.root)) {
        /* if VE is mounted -- umount first, to cleanup mount state */
        vps_umount(h, veid, res->fs.root, skip);
    }
```

I didn't do C for a while, so I'm not sure I'll be able to do something simple and safe.

I'm available to compile and test this until the openvz server should be production at the end of

next week... but I'll take the
time to do all the test we need before.

If some can help me on this issue.

thanks a lot

Yoann

Subject: Re: [vzctl] exec early script at start
Posted by [Yoann Moulin](#) on Tue, 08 Jan 2008 16:32:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

> Hi all,
>
> I'm working on a xen to openvz migration, and I still have some issue to
> fix. but some seem to be not possible yet.
>
> actually, on our diskless servers (xen and others), we are using
> nfsroot. And at start, in the initrd, we are using a script that make a
> connexion on the nfs boot server and remove the old root, create a new
> one, take the last system image build, and re apply configuration with
> cfengine tools.
>
> I'd like to do the same with openvz, but I didn't see in the source this
> possibility.
>
> it could be someting like in src/lib/env.c in the vps_start_custom
> function (in pseudo code) :

> I didn't do C for a while, so I'm not sure I'll be able to do something
> simple and safe.
>
> I'm available to compile and test this until the openvz server should be
> production at the end of next week... but I'll take the time to do all
> the test we need before.
>
> If some can help me on this issue.

I have patched env.c to be able to execute a script before starting a VE

I don't know if it is the best way to do that, but it work for what I need

regards,

Yoann

```

diff -r -u a/vzctl-3.0.22/include/vzerror.h b/vzctl-3.0.22/include/vzerror.h
--- a/vzctl-3.0.22/include/vzerror.h 2007-12-17 14:44:21.000000000 +0100
+++ b/vzctl-3.0.22/include/vzerror.h 2008-01-08 12:53:25.000000000 +0100
@@ -63,7 +63,7 @@
#define VZ_CANT_ADDIP 34
#define VZ_VALIDATE_ERROR 35
#define VZ_OVERCOMMIT_ERROR 36
-
+#define VZ_EARLY_SCRIPT_ERROR 37
/*****
Filesystem errors
*****/
diff -r -u a/vzctl-3.0.22/src/lib/env.c b/vzctl-3.0.22/src/lib/env.c
--- a/vzctl-3.0.22/src/lib/env.c 2007-12-17 14:44:21.000000000 +0100
+++ b/vzctl-3.0.22/src/lib/env.c 2008-01-08 17:18:14.786271752 +0100
@@ -545,12 +545,15 @@
{
    int wait_p[2];
    int err_p[2];
- int ret, err;
+ int ret, err, early_p;
    char buf[64];
    char *dist_name;
    struct sigaction act;
+ char *command;
+ char *early_script;
    vps_res *res = &param->res;
    dist_actions actions;
+     early_script = strdup("/etc/vz/early_script.sh");

    memset(&actions, 0, sizeof(actions));
    if (check_var(res->fs.root, "VE_ROOT is not set"))
@@ -559,6 +562,28 @@
    logger(-1, 0, "VE is already running");
    return VZ_VE_RUNNING;
}
+
+ if (early_script == NULL) {
+     logger(-1, 0, "memory allocation error for early_script");
+     return VZ_EARLY_SCRIPT_ERROR;
+ }
+
+ early_p = open(early_script, O_RDONLY);
+ if ( early_p != -1 ) {
+     logger(0, 0, "early_script.sh started");
+     command = (char*)malloc(3+strlen(early_script)+1+10);
+     sprintf(command, "sh %s %d", early_script, veid);

```

```

+ if ( system(command) == 0 ) {
+   logger(0, 0, "early_script.sh finished");
+ } else {
+   logger(-1, 0, "early_script.sh failed");
+   return VZ_EARLY_SCRIPT_ERROR;
+ }
+ free(command);
+ } else {
+   logger(0, 0, "early_script.sh does not exist or is not readable ");
+ }
+
+   if ((ret = check_ub(&res->ub)))
+     return ret;
+   dist_name = get_dist_name(&res->tpl);
@@ -661,6 +686,7 @@
+   close(wait_p[1]);
+   close(err_p[0]);
+   close(err_p[1]);
+ close(early_p);

+   return ret;
+ }

```

Subject: Re: [vzctl] exec early script at start
 Posted by [Yoann Moulin](#) on Mon, 14 Jan 2008 15:40:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

>> I didn't do C for a while, so I'm not sure I'll be able to do
 >> something simple and safe.
 >>
 >> I'm available to compile and test this until the openvz server should
 >> be production at the end of next week... but I'll take the time to do
 >> all the test we need before.
 >>
 >> If some can help me on this issue.
 >
 > I have patched env.c to be able to execute a script before starting a VE
 >
 > I don't know if it is the best way to do that, but it work for what I need

I had no feed back on this patch, is it will be applied ? or maybe nobody need this but me ?

I don't know if it's possible to do this better, or add a better security check, but as I said in my first mail, I don't developed in C for a while

thanks

Yoann

```
diff -r -u a/vzctl-3.0.22/include/vzerror.h b/vzctl-3.0.22/include/vzerror.h
--- a/vzctl-3.0.22/include/vzerror.h 2007-12-17 14:44:21.000000000 +0100
+++ b/vzctl-3.0.22/include/vzerror.h 2008-01-08 12:53:25.000000000 +0100
@@ -63,7 +63,7 @@
#define VZ_CANT_ADDIP 34
#define VZ_VALIDATE_ERROR 35
#define VZ_OVERCOMMIT_ERROR 36
-
+#define VZ_EARLY_SCRIPT_ERROR 37
/*****
Filesystem erros
*****/
diff -r -u a/vzctl-3.0.22/src/lib/env.c b/vzctl-3.0.22/src/lib/env.c
--- a/vzctl-3.0.22/src/lib/env.c 2007-12-17 14:44:21.000000000 +0100
+++ b/vzctl-3.0.22/src/lib/env.c 2008-01-08 17:18:14.786271752 +0100
@@ -545,12 +545,15 @@
{
    int wait_p[2];
    int err_p[2];
- int ret, err;
+ int ret, err, early_p;
    char buf[64];
    char *dist_name;
    struct sigaction act;
+ char *command;
+ char *early_script;
    vps_res *res = &param->res;
    dist_actions actions;
+     early_script = strdup("/etc/vz/early_script.sh");

    memset(&actions, 0, sizeof(actions));
    if (check_var(res->fs.root, "VE_ROOT is not set"))
@@ -559,6 +562,28 @@
    logger(-1, 0, "VE is already running");
    return VZ_VE_RUNNING;
}
+
+ if (early_script == NULL) {
+  logger(-1, 0, "memory allocation error for early_script");
+  return VZ_EARLY_SCRIPT_ERROR;
+ }
+
+ early_p = open(early_script, O_RDONLY);
```

```

+ if ( early_p != -1 ) {
+   logger(0, 0, "early_script.sh started");
+   command = (char*)malloc(3+strlen(early_script)+1+10);
+   sprintf(command,"sh %s %d",early_script,veid);
+   if ( system(command) == 0 ) {
+     logger(0, 0, "early_script.sh finished");
+   } else {
+     logger(-1, 0, "early_script.sh failed");
+     return VZ_EARLY_SCRIPT_ERROR;
+   }
+   free(command);
+ } else {
+   logger(0, 0, "early_script.sh does not exist or is not readable ");
+ }
+
+   if ((ret = check_ub(&res->ub)))
+     return ret;
+   dist_name = get_dist_name(&res->tmpl);
@@ -661,6 +686,7 @@
+   close(wait_p[1]);
+   close(err_p[0]);
+   close(err_p[1]);
+ close(early_p);

return ret;
}

```

Subject: Re: [vzctl] exec early script at start
 Posted by [kir](#) on Mon, 14 Jan 2008 16:38:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

Technically, if you want to do something before vzctl start executes,
 the way to go would be to write a simple shell wrapper which will do
 what's needed and then run vzctl. Something like this:

```

#!/bin/sh
ACTION=$1
VEID=$2

start_actions() {
    # put here all you need to do before vzctl start
}

stop_actions() {
    # same for stop
}

```

```

case $ACTION in
    start)
        start_actions $*
        ;;
    stop)
        stop_actions $*
        ;;
    # anything else you need here
esac

# Finally, exec vzctl
exec /usr/sbin/vzctl $*

```

Now, you just call the above script instead of calling vzctl directly. You can even call the script 'vzctl', if you either put it into a directory which is before /usr/sbin in your \$PATH, or rename "real" vzctl into something like vzctl.real.

Yoann Moulin wrote:

```

> Hi,
>
>>> I didn't do C for a while, so I'm not sure I'll be able to do
>>> something simple and safe.
>>>
>>> I'm available to compile and test this until the openvz server
>>> should be production at the end of next week... but I'll take the
>>> time to do all the test we need before.
>>>
>>> If some can help me on this issue.
>>
>> I have patched env.c to be able to execute a script before starting a VE
>>
>> I don't know if it is the best way to do that, but it work for what I
>> need
>
> I had no feed back on this patch, is it will be applied ? or maybe nobody
> need this but me ?
>
> I don't know if it's possible to do this better, or add a better
> security
> check, but as I said in my first mail, I don't developed in C for a while
>
> thanks
>
> Yoann
> -----
>

```

Kir Kolyshkin a écrit :

> Technically, if you want to do something before vzctl start executes,
> the way to go would be to write a simple shell wrapper which will do
> what's needed and then run vzctl. Something like this:

```
>
> #!/bin/sh
> ACTION=$1
> VEID=$2
>
> start_actions() {
>   # put here all you need to do before vzctl start
> }
>
> stop_actions() {
>   # same for stop
> }
>
> case $ACTION in
>   start)
>     start_actions $*
>     ;;
>   stop)
>     stop_actions $*
>     ;;
>   # anything else you need here
> esac
>
> # Finally, exec vzctl
> exec /usr/sbin/vzctl $*
>
> Now, you just call the above script instead of calling vzctl directly.
> You can even call the script 'vzctl', if you either put it into a
> directory which is before /usr/sbin in your $PATH, or rename "real"
> vzctl into something like vzctl.real.
```

it was my first id, but I thought it wasn't compatible with reboot action (I saw today that reboot is done by cron which call vzctl).

So it's necessary to call that script vzctl or change the script call by cron when a reboot is ask from a VE.

But; is that compatible with "vzctl restart" ? I meant 'restart' can't be call, script must do a stop, execute some action, then do a start, so with that script 'restart' option for vzctl will be deprecated .

that's why I've decided to patch env.c because it's was the most transparency way to do what I want, without change anything in how vzctl works if the 'early_script.sh' doesn't exist.

Yoann

Subject: Re: [vzctl] exec early script at start
Posted by [kir](#) on Wed, 16 Jan 2008 10:22:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

Yoann Moulin wrote:

> Kir Kolyshkin a écrit :

>> Technically, if you want to do something before vzctl start executes,
>> the way to go would be to write a simple shell wrapper which will do
>> what's needed and then run vzctl. Something like this:

>> <....skipped....>

> it was my first id, but I thought it wasn't compatible with reboot
> action (I saw today that reboot is done by cron which call vzctl).

In this case, rename vzctl to vzctl.real and put your wrapper script in place of vzctl.

>

> So it's necessary to call that script vzctl or change the script call
> by cron when a reboot is ask from a VE.

>

> But; is that compatible with "vzctl restart" ? I meant 'restart' can't
> be call, script must do a stop, execute some action, then do a start,
> so with that script 'restart' option for vzctl will be deprecated .

Right, here comes the problem.

>

> that's why I've decided to patch env.c because it's was the most
> transparency way to do what I want, without change anything in how
> vzctl works if the 'early_script.sh' doesn't exist.

Yeah, now I understand.

Have you tried using /etc/vz/conf/\$VEID.start script? There are also .stop, .mount and .umount. I'm sorry if it is not documented; will fix that.

Note that mount/umount scripts are both per-VE and global (global one is called vps.mount/vps.umount), while start/stop are only per-VE. This can easily be fixed: look at src/lib/env.c, functions vps_start_custom() and vps_stop() to see how start/stop scripts is called, then look at src/lib/fs.c, functions vps_mount() and vps_umount() to see how both global and per-VE mount/umount scripts are called. Now you can modify code in env.c to have global start/stop scripts as well. Patches are welcome.

PS the best way to do patches is to use git.

Subject: Re: [vzctl] exec early script at start
Posted by [Yoann Moulin](#) on Thu, 17 Jan 2008 14:41:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

> Have you tried using /etc/vz/conf/\$VEID.start script? There are also
> .stop, .mount and .umount. I'm sorry if it is not documented; will fix that.

> Note that mount/umount scripts are both per-VE and global (global one is
> called vps.mount/vps.umount), while start/stop are only per-VE. This can
> easily be fixed: look at src/lib/env.c, functions vps_start_custom()
> and vps_stop() to see how start/stop scripts is called, then look at
> src/lib/fs.c, functions vps_mount() and vps_umount() to see how both
> global and per-VE mount/umount scripts are called. Now you can modify
> code in env.c to have global start/stop scripts as well. Patches are
> welcome.

I have look, but start or mount script are call too late, as I said in my first message, I'd like to execute a script before mounting the rootfs for the VE, so maybe it can be done by adding a \$VEID.premount and be place just after the vps_is_run test.

I'm not very familiar with this syntaxes but I think it can be done similarly :

```
snprintf(buf, sizeof(buf), VPS_CONF_DIR "%d.%s", veid,
        START_PREFIX);
if (stat_file(buf)) {
    if (vps_exec_script(h, veid, res->fs.root, NULL, NULL,
        buf, NULL, 0))
    {
        ret = VZ_ACTIONSRIPT_ERROR;
        goto err;
    }
}
```

I'm investigate more into the code, and refresh some C knowledge to understand this

> PS the best way to do patches is to use git.

ok, I never use git before, but I will

Yoann

Subject: Re: [vzctl] exec early script at start
Posted by [Yoann Moulin](#) on Thu, 17 Jan 2008 16:07:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

> I'm not very familiar with this syntaxes but I think it can be done

```

> similarly :
>
> snprintf(buf, sizeof(buf), VPS_CONF_DIR "%d.%s", veid,
>     START_PREFIX);
> if (stat_file(buf)) {
>     if (vps_exec_script(h, veid, res->fs.root, NULL, NULL,
>         buf, NULL, 0))
>     {
>         ret = VZ_ACTIONSRIPT_ERROR;
>         goto err;
>     }
> }
>
> I'm investigate more into the code, and refresh some C knowledge to
> understand this

```

it's not possible to use vps_exec_script due to the script is execute inside the VE and the script I want to execute is here to prepare the root environnement by removing the old dir, populate the new dir and configure services with cfengine tools.

Yoann

Subject: Re: [vzctl] exec early script at start
 Posted by [Yoann Moulin](#) on Tue, 22 Apr 2008 13:35:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

> PS the best way to do patches is to use git.

in attachment, the patch generate with git

Regards

Yoann

```

diff --git a/include/vzerror.h b/include/vzerror.h
index 00d1c47..9d665d2 100644
--- a/include/vzerror.h
+++ b/include/vzerror.h
@@ -63,7 +63,7 @@
#define VZ_CANT_ADDIP 34
#define VZ_VALIDATE_ERROR 35
#define VZ_OVERCOMMIT_ERROR 36
-
+#define VZ_EARLY_SCRIPT_ERROR 37
/*****
Filesystem erros

```

```

*****/
diff --git a/src/lib/env.c b/src/lib/env.c
index 7411504..e1beac4 100644
--- a/src/lib/env.c
+++ b/src/lib/env.c
@@ -547,12 +547,15 @@ int vps_start_custom(vps_handler *h, envid_t veid, vps_param
*param,
{
    int wait_p[2];
    int err_p[2];
- int ret, err;
+ int ret, err, early_p;
    char buf[64];
    char *dist_name;
    struct sigaction act;
+ char *command;
+ char *early_script;
    vps_res *res = &param->res;
    dist_actions actions;
+     early_script = strdup("/etc/vz/early_script.sh");

    memset(&actions, 0, sizeof(actions));
    if (check_var(res->fs.root, "VE_ROOT is not set"))
@@ -561,6 +564,28 @@ int vps_start_custom(vps_handler *h, envid_t veid, vps_param *param,
    logger(-1, 0, "Container is already running");
    return VZ_VE_RUNNING;
}
+
+ if (early_script == NULL) {
+     logger(-1, 0, "memory allocation error for early_script");
+     return VZ_EARLY_SCRIPT_ERROR;
+ }
+
+ early_p = open(early_script, O_RDONLY);
+ if ( early_p != -1 ) {
+     logger(0, 0, "early_script.sh started");
+     command = (char*)malloc(3+strlen(early_script)+1+10);
+     sprintf(command, "sh %s %d", early_script, veid);
+     if ( system(command) == 0 ) {
+         logger(0, 0, "early_script.sh finished");
+     } else {
+         logger(-1, 0, "early_script.sh failed");
+         return VZ_EARLY_SCRIPT_ERROR;
+     }
+     free(command);
+ } else {
+     logger(0, 0, "early_script.sh does not exist or is not readable ");
+ }

```

```
+
  if ((ret = check_ub(&res->ub)))
    return ret;
  dist_name = get_dist_name(&res->tmpl);
@@ -663,6 +688,7 @@ err:
  close(wait_p[1]);
  close(err_p[0]);
  close(err_p[1]);
+ close(early_p);

  return ret;
}
```
