Hi Jeff,

This is src/lib/dist.c, because all such "setting" scripts are distro-dependent, and in
src/lib/vps_configure.c (because we used to call a VE a VPS).

Set of script to be run is determined by an /etc/vz/dist/$NAME.conf, where $NAME is taken from
the value of OSTEMPLATE parameter of VE config file (/etc/vz/conf/$VEID.conf). Note that the
value of OSTEMPLATE is taken literally first, then everything after and including the last '-' is
taken out and tried again, and so on until the config file is found. If a per-distro config file is not
found (say, due to bad OSTEMPLATE value), default (DIST_CONF_DEF defined in include/dist.h)
is used. See get_dist_conf_name() in src/lib/dist.c for details.

After the per-distro config file is found it it read and dist_actions struct is filled in. Now vzctl knows
which scripts to call. Names of scripts in file are relative to /etc/vz/dists/scripts directory.

Now, when a "set" action is run, appropriate scripts are called:
  src/vzctl-actions.c: set() calls
    src/lib/res.c vps_setup_res() calls
      src/lib/vps_configure.c vps_configure() calls
        specific action scripts one by one, using
          /src/lib/exec.c vps_exec_script() function

I suggest to use some code traversing tools such as vim + ctags to navigate all this, or gdb.

Now to your direct questions

Quote:Can I replace these shell scripts with any executable program to garner the same data as
long as my return values match up to what vzctl expects? Is there any documentation for this
interfacing?
Yes you can, if you place your stuff to under /etc/vz/conf/dists/scripts and make it executable.

Documentation describing all the scripts is in vzctl sources here (vzctl with those changes are not
yet released):
http://git.openvz.org/?p=vzctl;a=blob;f=etc/dists/distribution.conf-template

Quote:I modifed the redhat-add_ip.sh script to call #!/bin/bash -x to output bash debug lines, and
expected them to be either to STDOUT or /var/log/vzctl.log but I see that output in neither, so
where is it called and when?

you can do something like this:
#/bin/bash

Enable bash debugging
set -x

```
# Define the log file
LOGFILE=/tmp/my-$$.log
# Redirect both stdout and stderr to $LOGFILE
exec >$LOGFILE 2>&1

echo 1
```

If you only want to redirect stderr (and thus bash debugging messages) use the following exec instead:

```
exec 2>$LOGFILE
```

Then, see your $LOGFILE _inside_ a VE (because the script is run inside a VE to improve security).

Quote:Does the OpenVZ community consider it better to directly manipulate the container data under /vz/private/CTID or is their a prime imperative to use the built in OpenVZ provided tools?

It doesn't matter really, just make sure to not screw up mixing both ways

---