Compiling on LXC containers

This is not the recommended way to compile Batocera, as will become evident with all the workarounds you have to do to make this work. Refer to the compile Batocera page for the suggested methods.

Doing this will render it impossible to submit pull requests directly from this repository. If you choose to still proceed, you can use the LXC contained repository for testing things, and make the actual changes to an additional cloned local repository to make the pull request with.

If you have a dedicated compiling server then compiling on LXC containers, for example on a Proxmox host (which is being used for this LXC tutorial example), is a very convenient way to balance workloads and therefore get the best out of your hardware.

Preparations

For good security reasons LXC containers on Proxmox are running in unprivileged mode by default. It just keeps the container isolated from your Proxmox host and therefore from other different systems running on it. Nevertheless you have indeed the possibility to run an LXC container in privileged mode for whatever reason if you want to. When compiling Batocera on LXC containers be aware that both, unprivileged and privileged mode, need some minor preparations which will cover you from running into compile errors.

Unprivileged containers

Compiling Batocera on an unprivileged LXC container without preparation will make you run into the following compile error (multiple times, as the same variable within the following file covered is used multiple times dynamically while compiling):

```
docker Cannot change ownership to uid 71588, gid 10513: Invalid argument
```

To prevent these compile error(s) according to file permissions on `tar` operations cause by missing tar option flags you need to make a new branch and edit the following files from Batocera Github source as follows:

1. Make the following edits to the `./buildroot/Makefile` at ~line 457

   the `-c` option shows you the line numbers within then `nano` editor, eg: `nano -c ./buildroot/Makefile`

   ```
   TAR_OPTIONS = $(call qstrip,$(BR2_TAR_OPTIONS)) -xf
   ```

   to

```
TAR_OPTIONS = $(call qstrip,$(BR2_TAR_OPTIONS)) -xfo
```

2. Make the following edits to the `./buildroot/Config.in` in the `config BR2_TAR_OPTIONS` at ~line 160

   ```
   default ""
   ```

   to

   ```
   default "--no-same-owner"
   ```

3. Make the following edits to the `./package/batocera/emulators/wine/dxvk/dxvk.mk` file at ~line 13

   ```
   mkdir -p $(@D)/target && cd $(@D)/target && tar xf
   $(DL_DIR)/$(DXVK_DL_SUBDIR)/$(DXVK_SOURCE)
   ```

   to

   ```
   mkdir -p $(@D)/target && cd $(@D)/target && tar xfo
   $(DL_DIR)/$(DXVK_DL_SUBDIR)/$(DXVK_SOURCE)
   ```

4. Make the following edits to the `./package/batocera/emulators/wine/dxvk-nvapi/dxvk-nvapi.mk` file at ~line 13

   ```
   mkdir -p $(@D)/target && cd $(@D)/target && tar xf
   $(DL_DIR)/$(DXVK_DL_SUBDIR)/$(DXVK_SOURCE)
   ```

   to

   ```
   mkdir -p $(@D)/target && cd $(@D)/target && tar xfo
   $(DL_DIR)/$(DXVK_DL_SUBDIR)/$(DXVK_SOURCE)
   ```

5. Make sure all of the files have saved.

Now you should be able to compile without file permission errors caused by missing tar options flags.

Privileged containers

Using privileged containers is NOT recommended because of security reasons!

Compiling Batocera on a privileged LXC container without preparation will make you run into the following compile error right away from the start of compiling:

```
docker: Error response from daemon: AppArmor enabled on system but the
docker-default profile could not be loaded: running '/sbin/apparmor_parser
apparmor_parser -Kr /var/lib/docker/tmp/docker-default270540193' failed with
output: apparmor_parser: Unable to replace "docker-default". Permission
denied; attempted to load a profile while confined?
```

To prevent this compile error according to *AppArmor* being enabled you need to shutdown your LXC container and edit the according config file of your LXC container on the Proxmox host filesystem like this:

```
nano /etc/pve/lxc/<yourContainerID>.conf
```

Add the following new options:

```
features: nesting=1
lxc.apparmor.profile: unconfined
lxc.cap.drop:
```

Save the file, quit the editor and fire up your LXC container. Now you should be able to compile without *AppArmor* errors.

From:
https://wiki.batocera.org/ - **Batocera.linux - Wiki**

Permanent link:
https://wiki.batocera.org/compiling_lxc_containers?rev=1636470561

Last update: **2021/11/09 15:09**