University Göttingen
GWDG/AG-C
Niklas Bölter, Hendrik Nolte

Exercise 2 / October 17, 2023
HPC System Administration / WiSe 2022/23
20 Minutes Total

Exercise Introduction

Before attempting the exercises in this document please ensure that you have finished the last tutorial sheet. This is rather an optional follow up, then a mandatory stand alone sheet. If you have any problems booting your compute nodes you can also create a cluster manager from a snapshot which we have preconfigured for the node `n0`. You can then do the steps for `n1` and copy, what we have done for `n0`.
I also want to emphasize that the tasks on this sheet are much more open. You can/need/should also look at the docs `https://warewulf.org/docs/development/index.html` for further info when trying to do the exercises.

Contents

Task 1: Basic Interaction with Overlays (5 min)

I mentioned in the lecture, that overlays are stored `/var/lib/warewulf/overlays`. Have a look at this directory and compare it with the output of `wwctl overlays list`. What relationship do you see?
Diving a little bit deeper into overlays, lets create a new one:
`$ wwctl overlay create test`
You can check it with:
`$ wwctl overlay list test --all`
It should show nothing. Similarly you can check that a new directory has been created in `/var/lib/warewulf/overlays`.
Next, lets create a test file, e.g. `/etc/testfile`. In order to do that, we first have to create the directory `/etc`. Now, instead of me giving you the exact command which you need to run, lets try to figure the right command out on your own. For this `wwctl` offers `--help` on all levels. This means you can start with:
`$ wwctl --help`
and then find you way forward to:
`$ wwctl overlay --help`
And then build your `/etc` directory.
Once you have figured out to create `/etc`, control that it exists, using both `wwctl` and also do:
`$ ls /var/lib/warewulf/overlays/test`
The next step is create a file in `/etc`.
Go ahead and create `/etc/testfile`. If you use:

`$ wwctl overlay edit test /etc/testfile`
be "warned" that you will end up in vi. You can edit the file by pressing `i` (Now you can type the content of the file) and if you want to quit you have to press `Esc` and type `:x` and hit enter. If you are not comfortable with vi yet, you can also use nano by using the following commands: `$ yum install -y nano`
`$ EDITOR=nano wwctl overlay edit test /etc/testfile`

Task 2: Basics About Systemoverlays (5 min)

Before we continue with the new overlay `test`, which we just created, lets have a look at the node configuration again:
`$ wwctl node list --all`
you will see that the default system overlay is `wwinit`. You can have a closer look at it in `/var/lib/warewulf/overlays/wwinit`. Just have a look around again and try to understand what is going on here. The `init` file is probably a good place to start. You can just:
`$ cat init`
and see where it takes you.

Once you are done, you can set our new overlay as the system overlay, instead of `wwinit`. Don't forget to build the overlays, since you are modifying files, but the iPXE expects a cpio archive. Hint: If you don't know what to do, ask yourself: Where is the system overlay configured and how do change those settings?
Note:If you did the homework exercises from the first exercise sheet, you can do the following tasks as well: When you have that changed, you can reboot the node, for which you changed the systemoverlay, using the Openstack interface.
What do you expect?
What do you observe?

Task 3: Basics About Runtimeoverlays (5 min)

Before we continue, please revert the changes you made in the last task. Have a look again at your node. What is you runtime overlay?
You can have a look at the files again. Now set the same overlay `test` to the runtime overlay
Note:If you did the homework exercises from the first exercise sheet, you can do the following tasks as well: Reboot the node again.
What do you expect?
What do you observe?
What is the difference now compared to the case before where you used it as system overlay?
How would you go to make a custom overlay?

Task 4: Templating (5 min)

Before you continue with this task, please revert the changes from the last task.
Write a file to `/root` which you can later source on the node which defines your MAC address. Hin: `Hwaddr`. Try to find more network settings which you can template in that file. Remember, that your template files need a `.ww` suffix. If you need inspiration have a look into:
`$ cat /var/lib/warewulf/overlays/generic/etc/hosts.ww`

Optional Task 5: Building Containers Using Singularity/Apptainer (5 min)

This is a difficult **additional** task which will support your understanding in the topic.

We are looking at three different ways to build customized images. In the first exercise sheet, we have downloaded a prebuilt container, opened a shell within it and modified the image. That required a rebuild of the image afterwards. We will now use apptainer to build our own container image.
You can go to `https://github.com/hpcng/warewulf/tree/4.4.x/containers/Apptainer` and fetch a template recipe file. Read the recipe and add whatever software you like. Remember you can build the container using (as root):

```
dnf install epel-release
dnf install apptainer
apptainer build --sandbox /tmp/newroot /path/to/Singularity/recipe.def
wwctl container import /tmp/newroot containername
```

You should also sync the user/group ids of the passwd database between the cluster-manager and the container using the `container syncuser` subcommand of `wwctl`.

(Homework) You can now change the container of a node to the one you just created and reboot the node.

Optional Task 6: Building Containers Using a Chroot (5 min)

This is a difficult **additional** task which will support your understanding in the topic.

The third is a simple `chroot` environment. You can initialize a new root directory with:

```
yum install --releasever 8 --installroot --assumeyes /tmp/newroot2 basesystem bash \
    chkconfig coreutils e2fsprogs ethtool filesystem findutils \
    gawk grep initscripts iproute iputils net-tools nfs-utils pam \
    psmisc rsync sed setup shadow-utils rsyslog tzdata util-linux \
    words zlib tar less gzip which util-linux openssh-clients \
    openssh-server dhclient pciutils vim-minimal shadow-utils \
    strace cronie crontabs cpio wget rocky-release ipmitool yum \
    NetworkManager
```

Once you are done, you can try to modify it, either just by editing the directory, or by first doing a `chroot` command. Once you are done, you can:
`$ sudo wwctl container import /tmp/newroot2 containername`
You should also sync the user/group ids of the passwd database between the cluster-manager and the container using the `container syncuser` subcommand of `wwctl`.

(Homework) You can now change the container of a node to the one you just created and reboot the node.