THE SYMLINK RANCH

AN UNPRIVILEGED OVERLAYFS FOR ALL DISTRIBUTIONS

AUTHOR : ELEXIS ANNASTASIA-DAVINA PANAS MENTORS : REID PRIEDHORSKY & JORDAN OGAS

THE UNIVERSITY OF NEW MEXICO

Los Alamos NATIONAL LABORATORY EST. 1943

SUMMARY

HPC supercomputers need more flexibility to enable more science capability to its users. A container is a process that can accomplish this goal

BACKGROUND

Users need read-only images for better image distribution but need read-write images to inject files such as shared libraries for nVidia GPUs and proprietary interconnects.

THE PROBLEM

The best supported way to enable the write permission is to use overlayfs, which lets you overlay a tmpfs on top of a read-only image. This serves those who are superusers however, most distributions do not allow overlayfs in unprivileged conditions.

THE PROTOTYPE

With a Symlink Ranch we are able to enable the write permission to a read-only image in a completely unprivileged environment for any distribution.

WHAT DOES THIS MEAN?

The Symlink Ranch solves the problem of writing to read-only images in an unprivileged environment. For example, in Charliecloud we are now able to inject files into a mounted read-only squash file system with the help of a Symlink Ranch.

This unprivileged solution allows for easier distribution of read-only images at a low cost to HPC which gives users more tools for image distribution on our supercomputers.

Getting Ready to Ranch

I want to add the file 'henry' to the read-only image barn1.

Ask about the C implementation!

```
$ unshare -mU -r bash
# ls -lhR
/data/flying-lex:
drwxr-sr-x 1 reidpr root 32 Jun 28 11:19 barn1
drwxr-sr-x 1 reidpr root  0 Jul  2 12:46 barn2

/data/flying-lex/barn1:
-rw-r--r-- 1 reidpr root 0 Jun 28 11:18 hereford
-rw-r--r-- 1 reidpr root 0 Jun 28 11:18 holstein
/data/flying-lex/barn2:
```

Step One – Bind Mount to Preserve

Bind mount to a second image 'barn2'.

barn1 and barn2 share the same Inode #.

```
# mount -o bind /data/flying-lex/barn1 /data/flying-lex/barn2
#findmnt -nRM /data -o TARGET,SOURCE
/data                    /dev/sdb1
├─/data/docker/btrfs     /dev/sdb1[/docker/btrfs]
└─/data/flying-lex/barn2 /dev/sdb1[/flying-lex/barn1]
```

```
# stat -c %n' '%i' '%F' '%A barn1 barn2
barn1 110062 directory drwxr-sr-x
barn2 110068 directory drwxr-sr-x

# mount -o bind /data/flying-lex/barn1 /data/flying-lex/barn2

# stat -c %n' '%i' '%F' '%A barn1 barn2
barn1 110062 directory drwxr-sr-x
barn2 110062 directory drwxr-sr-x
```

Step Two – Mount tmpfs to Write

This overlays a tmpfs onto the barn1 image.

```
# mount -t tmpfs none /data/flying-lex/barn1
#  findmnt -nRM /data -o TARGET,SOURCE
/data                    /dev/sdb1
├─/data/docker/btrfs     /dev/sdb1[/docker/btrfs]
├─/data/flying-lex/barn2 /dev/sdb1[/flying-lex/barn1]
└─/data/flying-lex/barn1 none
```

barn1 has a different Inode #.

```
# stat -c %n' '%i' '%F' '%A barn1 barn2
barn1 419745 directory drwxrwxrwt
barn2 110062 directory drwxr-sr-x
```

Step Three – Symlink to Restore

```
# ln -s /data/flying-lex/barn2/hereford /data/flying-lex/barn1/hereford
# ln -s /data/flying-lex/barn2/holstein /data/flying-lex/barn1/holstein
# ls -lhR
.:
./barn1:
lrwxrwxrwx 1 root root 31 Jul  2 13:04 hereford -> /data/flying-lex/barn2/hereford
lrwxrwxrwx 1 root root 31 Jul  2 13:05 holstein -> /data/flying-lex/barn2/holstein
./barn2:
-rw-r--r-- 1 nobody nogroup 0 Jun 28 11:18 hereford
-rw-r--r-- 1 nobody nogroup 0 Jun 28 11:18 holstein
```

Step Four – Write on your Image

```
# touch henry
# ls -lhR
.:
drwxrwxrwt 2 root   root    100 Jul  2 13:05 barn1
drwxr-sr-x 1 nobody nogroup  32 Jun 28 11:19 barn2
./barn1:
-rw-r--r-- 1 root root  0 Jul  2 13:05 henry
lrwxrwxrwx 1 root root 31 Jul  2 13:04 hereford -> /data/flying-lex/barn2/hereford
lrwxrwxrwx 1 root root 31 Jul  2 13:05 holstein -> /data/flying-lex/barn2/holstein
./barn2:
-rw-r--r-- 1 nobody nogroup 0 Jun 28 11:18 hereford
-rw-r--r-- 1 nobody nogroup 0 Jun 28 11:18 holstein
```

Add the file 'henry'.

Symlinks make your content appear where you want it to, while mount commands help to enable the write permission for the read-only image.

Now you can write to a read-only image!

LA-UR-19-27119