Updates and Innovations with the Apptainer Platform

Forrest Burt - Solutions Architect, CIQ
February 3, 2024
FOSDEM 2024
Brussels, Belgium
Developments in Apptainer over the past few years

- Leveraging the User Namespace
- New recommendations for containerized MPI
- Increased adoption of ORAS (for Software Supply Chain)
Leveraging the User Namespace

Namespace types
The following table shows the namespace types available on Linux. The second column of the table shows the various APIs. The third column identifies the manual page that provides details on the namespace type. The fourth column displays the namespace type.

<table>
<thead>
<tr>
<th>Namespace</th>
<th>Flag</th>
<th>Page</th>
<th>Isolates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cgroup</td>
<td>CLONE_NEWCGROUP</td>
<td>cgroup_namespaces(7)</td>
<td>Cgroup root directory</td>
</tr>
<tr>
<td>IPC</td>
<td>CLONE_NEWIPC</td>
<td>ipc_namespaces(7)</td>
<td>System V IPC, POSIX message queues</td>
</tr>
<tr>
<td>Network</td>
<td>CLONE_NEWNET</td>
<td>network_namespaces(7)</td>
<td>Network devices, stacks, ports, etc.</td>
</tr>
<tr>
<td>Mount</td>
<td>CLONE_NEWNS</td>
<td>mount_namespaces(7)</td>
<td>Mount points</td>
</tr>
<tr>
<td>PID</td>
<td>CLONE_NEWPID</td>
<td>pid_namespaces(7)</td>
<td>Process IDs</td>
</tr>
<tr>
<td>Time</td>
<td>CLONE_NEWTIME</td>
<td>time_namespaces(7)</td>
<td>Boot and monotonic clocks</td>
</tr>
<tr>
<td>User</td>
<td>CLONE_NEWUSER</td>
<td>user_namespaces(7)</td>
<td>User and group IDs</td>
</tr>
<tr>
<td>UTS</td>
<td>CLONE_NEWUTS</td>
<td>uts_namespaces(7)</td>
<td>Hostname and NIS domain name</td>
</tr>
</tbody>
</table>

The namespaces API
As well as various /proc files described below, the namespaces API includes the following system calls...
Leveraging the User Namespace (for --fakeroot)

```
[demouser@ciqbox ~]$ id
uid=1001(demouser) gid=1001(demouser) groups=1001(demouser) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023

[demouser@ciqbox ~]$ pwd
/home/demouser

[demouser@ciqbox ~]$ touch foo

[demouser@ciqbox ~]$ ll
total 0
-rw-r--r-- 1 demouser demouser 0 Jul 27 14:55 foo

[demouser@ciqbox ~]$ 
```
Leveraging the User Namespace (for --fakeroot)

```
[demouser@ciqbox ~]$ apptainer shell --fakeroot docker://alpine
Apptainer> id
uid=0(root) gid=0(root) groups=0(root)
Apptainer> pwd
/root
Apptainer> ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 14:55 foo
Apptainer> touch bar
Apptainer> ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 14:58 bar
-rw-r--r-- 1 root root 0 Jul 27 14:55 foo
Apptainer> exit

[demouser@ciqbox ~]$ ll
total 0
-rw-r--r-- 1 demouser demouser 0 Jul 27 14:58 bar
-rw-r--r-- 1 demouser demouser 0 Jul 27 14:55 foo

[demouser@ciqbox ~]$
```
Leveraging the User Namespace (for **build**)
Leveraging the User Namespace (for build)
Leveraging the User Namespace (for installation)

- Default Apptainer installation is now unprivileged
- Squashfuse is used to mount the squashfs file system
- Apptainer enters a new User Namespace and then creates a new Mount Namespace to present the new root filesystem to processes
Leveraging the User Namespace (for installation)
Leveraging the User Namespace (for installation)

**Important Note:**

Because users can install in their own space they can manage their own configuration. This means that **things like ECL may be rendered invalid!**

Must disable User Namespace in OS if this is a problem.
New recommendations for MPI jobs

- Containerized MPI jobs can present difficulties
  - wire-up
  - fabric adapter

- Historical approaches are difficult and lack portability

- PMI and libfabric are recommended to increase container portability

- Container-friendly package managers (like Spack) can reduce difficulty
New recommendations for MPI jobs

Review of the wire-up problem
New recommendations for MPI jobs

Review of the wire-up problem

```bash
mpiexec ... proc
```

```
ssh
```
New recommendations for MPI jobs

Review of the wire-up problem
New recommendations for MPI jobs

Solving the wire-up problem with PMI

```
srun --mpi=pmi2 ... apptainer ... img.sif proc
```
New recommendations for MPI jobs
Links to detailed resources

- Blog post with detailed (copy/paste-able) scripts
- Jonathon Anderson presenting this work at the 2023 HPC-AI Advisory council meeting at Stanford
- Dave Godlove and Jonathon Anderson discussing this work in a CIQ Webinar
New recommendations for MPI jobs

Review of the fabric adapter problem
New recommendations for MPI jobs

Solving the fabric adapter problem with libfabric

compute node

MPI enabled application

MPI implementation

libfabric

kernel (with module(s) for fabric adapter)

hardware fabric adapter
ORAS for Software Supply Chain

● The OCI Registry As Storage (ORAS) protocol allows native SIF files to be stored on OCI registries (like Docker Hub)

● You can use advanced features like signing, verifying, and encryption without giving up the convenience of OCI registries!
ORAS for Software Supply Chain

Overview

These are signed SIF images of Rocky Linux built from the upstream mirrors (as opposed to building from existing containers). These can be pulled and verified as known good base containers to help with software supply chain concerns.

Docker Pull Command

docker pull godlovedc/rockylinux
ORAS for Software Supply Chain

```
[demouser@ciqbox ~]$ apptainer pulloras://docker.io/godlovedc/rockylinux:8
INFO: Downloading oras image

[demouser@ciqbox ~]$ apptainer verify rockylinux_8.sif
INFO: Verifying image with PGP key material
[REMOTE] Signing entity: David Godlove (production key) <davidgodlove@gmail.com>
[REMOTE] Fingerprint: B7761495F83E6BF7686CA5F0C1A7D02200787921

Objects verified:
ID | GROUP | LINK    | TYPE
-----------------------------
1  | 1      | NONE    | Def.FILE
2  | 1      | NONE    | JSON.Generic
3  | 1      | NONE    | FS
INFO: Verified signature(s) from image 'rockylinux_8.sif'

[demouser@ciqbox ~]$ 
```
ORAS for Software Supply Chain

[demouser@ciqbox ~]$ apptainer inspect --deffile rockylinux_8.sif
BootStrap: yum
OSVersion: 8
MirrorURL: http://dl.rockylinux.org/pub/rocky/%{OSVERSION}/BaseOS/x86_64/os/
Include: dnf

%labels
  Author: davidgodlove@gmail.com

%post
  dnf -y update
  dnf install -y epel-release file

%environment
  LC_ALL=C

[demouser@ciqbox ~]$
ORAS for Software Supply Chain

Bootstrap: oras
From: docker.io/godlovedc/rockylinux:8
Fingerprints: B7761495F83E6BF7686CA5F0C1A7D02200787921

%post
   echo 'only install if signed and verified!'
ORAS for Software Supply Chain

```
[demouser@ciqbox ~]$ apptainer build sigexample.sif sigexample.def
WARNING: 'nodev' mount option set on /tmp, it could be a source of failure during build process
INFO:   Starting build...
INFO:   Downloading oras image
INFO:   Checking bootstrap image verifies with fingerprint(s): [B7761495F83E6BF7686CA5F0C1A7D02200787921]
INFO:   Running post scriptlet
+ echo 'only install if signed and verified!'
+ only install if signed and verified!
INFO:   Creating SIF file...
INFO:   Build complete: sigexample.sif
```

```
ORAS for Software Supply Chain

[demouser@ciqbox ~]$ aptainer build sigexample.sif sigexample.def
WARNING: 'nodev' mount option set on /tmp, it could be a source of failure during build process
INFO:   Starting build...
INFO:   Downloading oras image
INFO:   Checking bootstrap image verifies with fingerprint(s): [B7761495F83E6BF7686CA5F0C1A7D82200787922]
FATAL:  While performing build: conveyor failed to get: while checking fingerprint: image not signed by required entities

[ demouser@ciqbox ~ ]$
Developments in Apptainer over the past few years

- Leveraging the User Namespace
- New recommendations for containerized MPI
- Increased adoption of ORAS (for Software Supply Chain)
Get involved!

- Website: https://apptainer.org/
- Get Started: https://apptainer.org/get-started/
- GitHub: https://github.com/apptainer/apptainer
- Community Slack, mailing list, etc links: https://apptainer.org/support/
- User docs: https://apptainer.org/docs/user/main/
- Admin docs: https://apptainer.org/docs/admin/main/