Containers in HPC
(the slides are available under “Presentation Materials” in the above page)
Date: January 23, 2019
Presented by: Shane Canon (LBNL/NERSC)

Q: You may have planned to, but hopefully you'll address the recently discovered RunC vulnerability: https://www.zdnet.com/article/doomsday-docker-security-hole-uncovered/

A: I didn’t discuss it today. This doesn’t impact most of the HPC runtimes but does impact systems that are used to host services on a shared platform and use a Docker-based infrastructure. For example, we had to update our Spin system at NERSC to address this vulnerability.

Q: How are containers combined/mixed and versioned on docker (see Git: forks and merges)?

A: Docker images aren’t merged and forked quite like Git. You can use base images and then modify from that which is a little like a fork or a branch. Also Docker images can have version tags. For example the image auser/image:v1 would have a version tag of “v1”.

Q: So the purpose of container is to create some virtual machines on the laptop for development? Are you using LXC?

A: I discussed this some in the session. First I would say the purpose of containers spans a very wide space, but the goal is create a well defined, encapsulated environment to run applications or services. This could include iterative development but could be the actual application to be run too. I wouldn’t think of it as a virtual machine since it is really just a process running in some semi-isolated, defined environment. It is much less overhead than a VM. LXC is one way to run containers, but is not the most common these days. Docker is by far the most popular.

Q: How are compiler versions being handled? That is always a problem.

A: Answered in the discussion, but compilers are just another thing that can be installed in the container environment. So the developer has total control over what compiler version is installed and used. This provides much greater flexibility and allows the developer to decide when is the right time to make a change. This shows one of the big advantages of containers.

Q: Does an image tied to some hardware architecture?

A: Yes. The generated image will have binary code in it that is for some targeted architecture. So containers can't be used to move between say x86 and PowerPC or make a code work on GPU. However, the Dockerfile used to build an image could potentially be used to build an image for different architectures depending on how specialized the "recipe" is.

Q: Can you run an mpi program on multiple nodes?

A: Yes ;-) (Addressed during the seminar)

Q: How are licenses of commercial software/compilers/libraries handled?

A: Good question. You can install commercial software in an image but you would still need to consider any restrictions placed on the software. At NERSC we run a private registry to allow users to store private images that could include commercial software. We also have recipes on how use the Intel Compilers to build images. This recipes require the user to create a tunnel to the NERSC license serve to run the compilers. So it is a bit complicated. My hope is that, in the future, vendors will provide easier solutions for these scenarios.

https://docs.nersc.gov/development/shifter/how-to-use/programming/shifter/intel/programming/shifter/intel/

Q: Can you chain two tools from different containers to process data in a row?

A: Yes. You can do things like pipes between containers or you can use volume mounts. For example….

```
docker run -v /scratch/user:/data image1  \
     tool1 /data/input1 /data/output1
docker run -v /scratch/user:/data image2 \
     tool2 /data/output1 /data/outpu2
```

...

Q: Do you do any form of container image scanning for security?

A: We currently do not for containers that run on HPC. The logic being that the container has no special privileges so the user could already, in theory, pull down software and run broken or insecure applications already. So the fundamental risk hasn't changed. However, we do run a container-based system for services that are connected to the broader internet. For those we are scanning to make sure they are okay. Also, (back to the HPC context) Shifter does have hooks so that a site could integrate scanning before an image is made available for execution. So if a site had more aggressive security posture, there are mechanisms.

Q: So Shifter only works with slurm, not Sun Grid engine or PBS?

A: Answered during discussion, but Shifter can work with these too and the batch system integration is optional. It mainly enhances scaling in certain circumstances and can provide a better user experience.

Q: Can you mount an ISO images in container?

A: Not by default. The ISO image would already need to be mounted or extracted.

Q: Reproducibility is always touted as one of the big advantages of containers. But as soon as you introduce the concept of dynamic libraries, and commonly you swap libs in and out in an HPC environment, then you lose all reproducibility. Right?

A: Mounting in the appropriate dynamic libraries for the system does detract from the reproducibility in the absolute sense. Generally we are only mapping in the libraries that are needed to take advantage of the interconnect hardware and the user can disable this (which would hurt performance). We felt like this was the best balance. In general though containers are a huge improvement since most of the pieces do not change. For non-MPI applications it is basically not modified at all.

Q: What would a Dockerfile look like for a application which runs on Cori and requires PetSc for example?

A: It depends on how you wanted to install PetSc but the basic recipe would be…

```
FROM centos:7

RUN yum -y install make gcc ….

RUN wget <petsc>-version.tgz && \
   tar xzf petsc-version.tgz && \
    cd petsc-version && ./configure <flags> && make && make
install

ADD . /myapp
RUN \
    cd /myapp && ./configure && make && make install
```

Q: Does Shifter integrate well also w/ Cobalt (in addition to Slurm)?

A: See the answer above. We haven't with Cobalt, but you don't have to integrate Shifter with the scheduler to use it. Integration just helps in some circumstances with performance and scaling.

Q: I want to pull images and run directly from within my SBATCH script from my own private registry. Can I do this from the compute nodes at Oak Ridge and NERSC? I.e are the compute nodes accessible to the external network to allow this, or must i have a wrapper that does this from login first before executing the sbatch file? Intent is to use singularity to pull and run images that contain entire application stack.

A: For Shifter, the image gateway sits outside the system and then converts and places it on the parallel file system. So this would work at NERSC. I'm not sure about ORNL.

Q. With singularity, the mpi version in my image needs to be the same as the host environment MPI version?

A: It depends on the MPI version. In general you need ABI compatibility. So if the MPI install is ABI compatible with the available libraries for the running system, things should work. We have run the same image at NERSC (using Shifter) over several years.

Q. Is there support for running stuff on the cloud?

A: In general, on the cloud I would just run Docker since you don't have to worry about the security and integration issues Shifter and the other HPC runtimes were designed to address. So, yes, you can run containers on the cloud.

Q. How is the performance of multi-node MPI codes running within shifter vs outside shifter ?

A: In all the cases I have seen, Shifter is as fast or faster. If it is slower it is probably because of some odd mistake in the image build or execution.

C. Just wanted to point out that containers aren't the only way to solve the scalable loading problem. You can do this on non-containerized Python (and other) applications with Spindle: https://github.com/hpc/spindle

Yes. I generally try to mention this. However, I think containers/Shifter are nice because they help with this and provide a nice reproducible artifact.

Q. Any comments on GPU support?

A: Good point. I should have addressed this. Other sites are using containers with GPUs. So the short answer is, yes you can use containers with GPUs. I'm hoping NERSC will get more direct experience with this very soon.

Q. What about PBS? Does Shifter work with PBS?

A: See above. There are examples. But the batch integration is optional. So you could still use Shifter (and the other HPC runtimes) without integrating into the batch system. For Shifter, the integration mostly helps with scaling.