Containers

DOSAR

Original slides by: Rob Quick <rquick@iu.edu>
Follow Along at:

Containers
What are containers?

• Operating System Level Virtualization
  – Lightweight, providing the minimal level of overhead for the application to function properly.
  – Super minimalist VMs
  – No Hypervisor
  – Abstracts away the operating system and hardware
  – Share the OS Kernel with other containers
  – Container size is very small and therefore quick and easy to provision
How do they differ from VMs?

Virtual Machines

- App 1
- Bins/Libs
- Guest OS
- Hypervisor
- Host OS
- Server

Containers

- App 1
- Bins/Libs
- Container Engine
- Host OS
- Server

- App 2
- Bins/Libs
More differences…

- **Size**
  - Containers are usually 10s of MB
  - VMs can be several GB
- **Shared hypervisor vs. shared kernel**
- **VMs have their own kernels so a deeper level of isolation**
- **Containers virtualize the OS while VMs virtualize the hardware**
Container Advantages

- Size
- Less resource intensive
- Quick provisioning
- Easy allocation of resources
- Quicker development cycles
- Cost effective
- Very good for microservices
Container Disadvantages

• Security – shared kernel with root access
• Less flexibility in OS
• Networking can be tricky
  – Properly configuring sufficient networking resources is challenging
Container Software

- Docker
- Singularity
- LXC, LXD
- Solaris Zones
- RKT
- BSD Jails
- chroot
Questions?

- Feel free to ask us questions now or later:
  Horst Severini  severini@ou.edu
  Pat Skubic   pskubic@ou.edu
  Julia Gray julia.ann.gray@gmail.com

Exercises start here:


Presentations are also available from this URL.