TAMING CONTAINER FEARS

Understanding the risk and reward

Scott McCarty
Senior Strategist, Containers, Red Hat
04/10/2016
AGENDA
What we’ll cover

The Reward
Why use containers

The Risk
How containers share a kernel

Tame Your Fear
How we can mitigate the risk

Discussion
Questions are good
THE REWARD
Load Applications at the Factory, not the Dock
The Problem

Applications require complicated collaboration during installation and integration every time they are deployed.

*Image: Manually Loading Ships 1921*
What About Virtualization?

Cargo holds help, but you still have to load the ship manually.
What about Configuration Management

Alone, it’s just, better boxes, bags, barrels, crates and forklifts
The solution

Containerize

Adopting a container strategy will allow applications to be easily shared and deployed.
The Journey

It’s definitely a journey....

Evaluate Technology  Experiment  Quick Win  Inventory Applications  Determine Technology  Containerize
CONTAINERS FOR THE ENTERPRISE

- Deliver apps faster
- Deploy & manage at scale
- Comprehensive security
- Unified environment
THE RISK
CONTAINERS DON’T CONTAIN

Dan Walsh (my shirt is dedicated to you)

Move the kernel around or move the user space around

- Fancy processes
- Breaking the OS in two pieces
- All containers share a kernel
- Root only exploits can be ba’aa’a’ad
Namespaces allow data structures to be virtualized

- User identifiers
- Group identifiers
- Process identifiers
STARTING PROCESSES

Processes can be started with exec(), fork(), or clone()

containers-deep-dive/containers201/demo-execve.sh
containers-deep-dive/containers201/demo-clone.sh
DEEPER NAMESPACES

It’s still a Linux operating system...

All kernel/system call rules apply

- Mount Namespace
- Virtual Filesystem
- Filesystem Driver
- Block Storage Driver
MORE THAN JUST NAMESPACE
Now, try to get out of this!!!

Clone()  cgroups
SELinux  SECCOMP
“What happens if root escalates to root?”

Josh Bressers said this to me, and I was like whaaaaaaaat?
(in appropriately high pitched voice)...
TAME YOUR FEAR
The Tenancy Scale

- Process
- Container
- Virtual Server
- Physical Server
- Rack
- Data Center
MITIGATE RISKS
Let’s see if we can tame the fear a bit...

Container technical controls
- Limit Root Access
- **SECCOMP**
- sVirt
- Read Only Containers
- Audit Data Access
- Drop Privileges
- Prevent New Privileges
RUNNING A SECURE CONTAINER

Showing a few of the technical controls in action

taming-container-fears/demo-hardened.sh
Conclusions

Yes, I am bringing it back....

- There is an amazing business benefit to containers
- Linux Containers share a kernel
- They can be locked down beyond what is convenient with normal process (in VMs on on bare metal)
Call to Action

Learn more. Ask questions.

- Container Defense in Depth: Wednesday @ 11:00
- Migrating Existing Applications: Wednesday @ 16:40
- A Practical Introduction to Docker: http://red.ht/2bPpZu9
- A Practical Introduction to Docker Terminology: http://red.ht/2bPpZu9
- Architecting Containers: http://red.ht/2aXjVJF
- Runc Tutorial: http://red.ht/2doofq4
THANK YOU