redhat.

TL;DR NIST Container Security Standards

Elsie Phillips
5/4/18

Slides:

First things first....

Agenda

1. Intro: What are we trying to accomplish/Am I in the right talk?
2. Let’s get on the same page: terminology and such
3. Security and containers
4. Recommendations

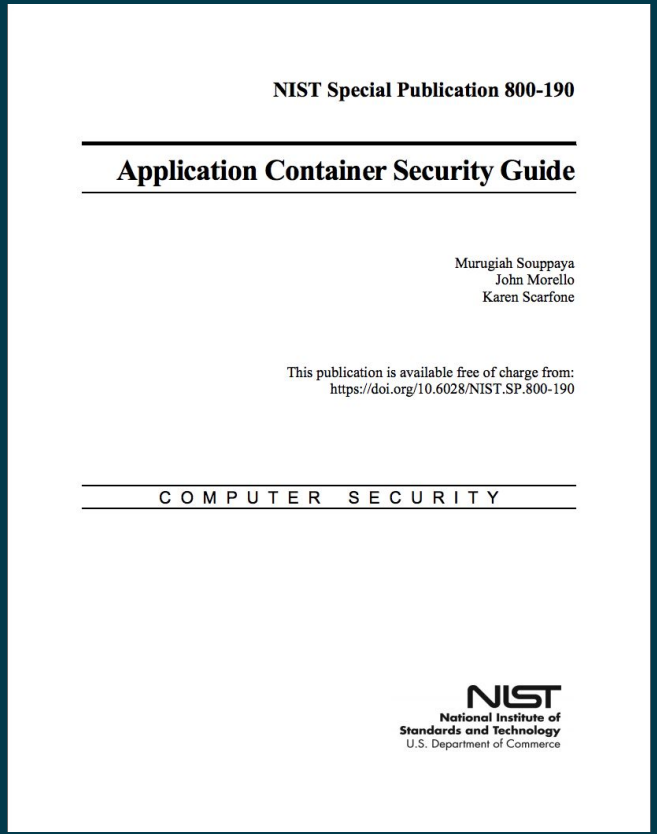
About me + this talk

Quick Intro

- Red Hat, formerly CoreOS
- Contrib-X Chair
- Got my start in OSS working at the OSU Open Source Lab
- Live in Berkeley, CA
- Directionally challenged
- Twitter: @elsiephilly

The Report

https://goo.gl/MDkExr



NIST Special Publication 800-190

Application Container Security Guide

Murugiah Souppaya
John Morello
Karen Scarfone

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-190

COMPUTER SECURITY

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

Report Basics

Click to add subtitle

- Who is this report aimed at?
- What will we be covering?
- What will we *not* be covering?

Terminology

Virtualization

Application Virtualization

Immutability

Container Runtime

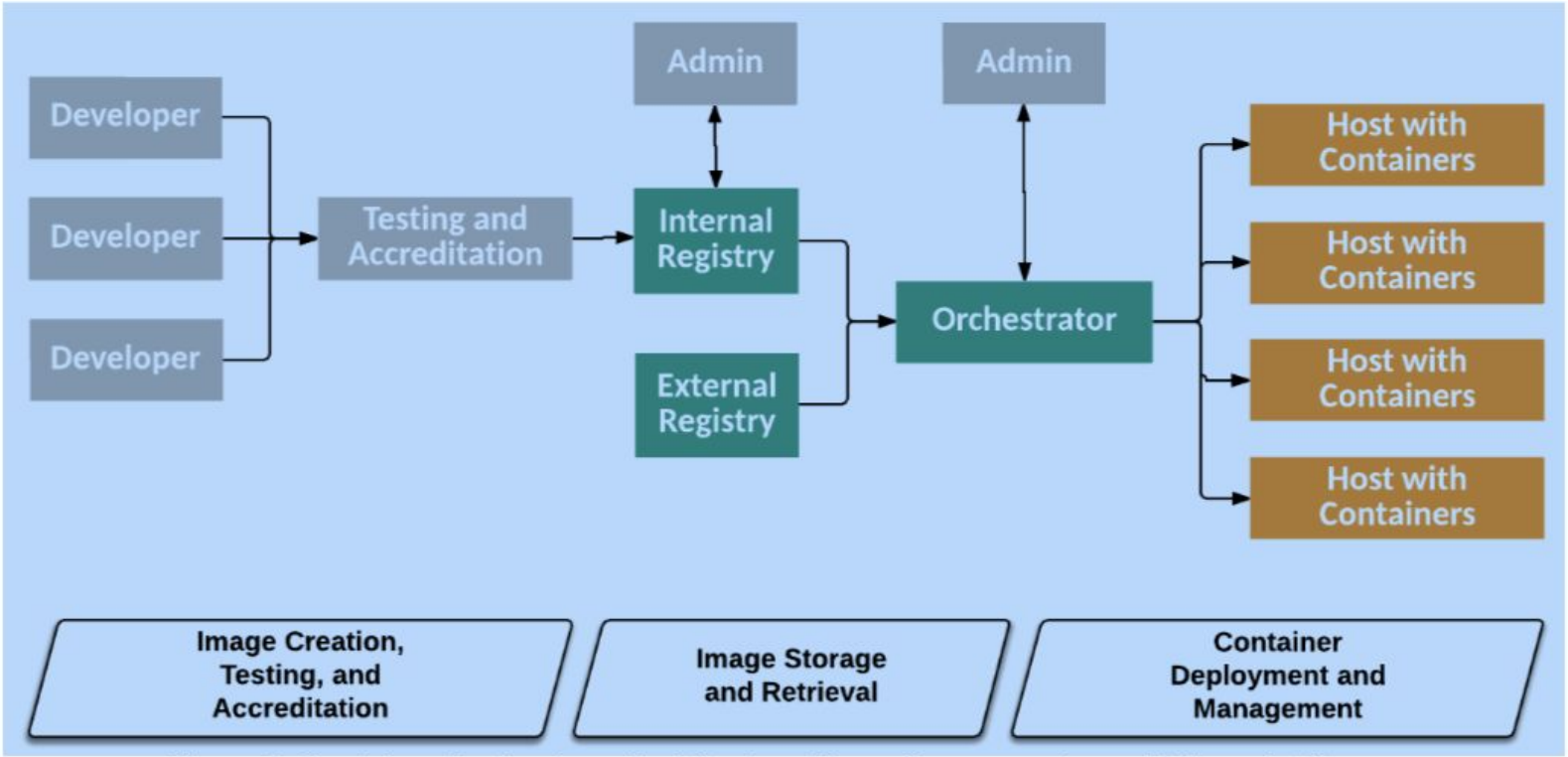


Figure 3: Container Technology Architecture Tiers, Components, and Lifecycle Phases

Major Risks For Core Components of Container Technologies

Image Risks

Image Vulnerabilities

Image Configuration Defects

Embedded Malware

Embedded Clear Text Secrets

Use of untrusted images

Registry Risks

Insecure connections to registries

Insufficient authentication and authorization restrictions

Orchestrator Risks

Unrestricted Admin and Unauthorized Access

Poorly separated inter-container network traffic

Orchestrator Node Trust

Container Risks

Vulnerabilities within the runtime

App Vulnerabilities

Rogue containers

Host OS component vulnerabilities

Shared Kernel

Improper user access rights

Host OS file system tampering

Recommendations

Image Vulnerabilities Countermeasures

- Use vulnerability management tools that are specifically designed for containers
- Adopt tools and processes that ensure compliance with secure configuration best practices
- Set-up embedded malware monitoring on all images
- Store secrets outside of images
- Establish a set of trusted images and only permit these images to be run in your environments

Registry Vulnerabilities Countermeasures

- Only connect to registries over secure encrypted channels
- Get rid of old images or tag the newest ones
- Authentication mandatory to access a registry with sensitive content

Orchestrator Vulnerabilities Countermeasures

- Limit administrative access
- Configure to separate network traffic by sensitivity level
- Isolate deployments to specific hosts by sensitivity levels
- Configure orchestration platform to create a secure environment for all apps they run

Container Vulnerabilities Countermeasures

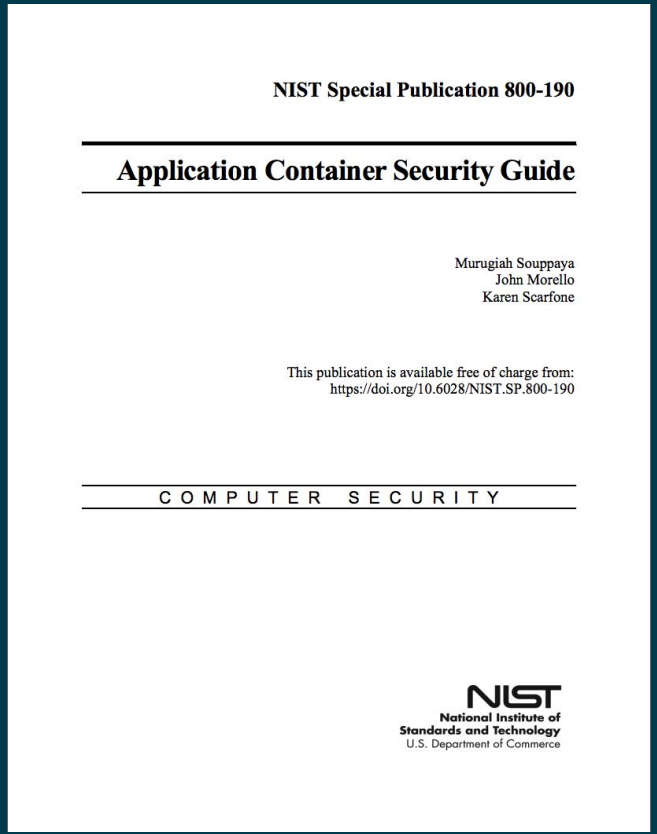
- Continually monitor runtime for vulnerabilities and quickly fix if found
- Automate compliance of container runtime configuration standards
- Container should only be run with their root in read-only mode
- Have separate environments for dev, test, prod, each w/ RBAC for container deployment and management

Host OS Vulnerabilities Countermeasures

- Use a container specific OS
- Don’t mix containerized and non containerized workloads on the same host
- Audit all authentication to the OS, monitor login anomalies, and privileged operations logged
- Run containers with the minimal set of file system permission required

The Report

https://goo.gl/MDkExr



NIST Special Publication 800-190

Application Container Security Guide

Murugiah Souppaya
John Morello
Karen Scarfone

This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-190

COMPUTER SECURITY

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

redhat.

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos