CLOUD NATIVE
COMPUTING FOUNDATION

The Do's and Don't for Securing Container and Cloud Native Technologies

Ty Sbano

Cloud Chief Information Security Officer - Sisense

Kavya Pearlman

Cybersecurity Strategist - Wallarm


KubeCon
CloudNativeCon
North America 2019
Nov. 18 - 21, 2019
San Diego, CA
kubecon.io

Kubernetes Forums

Sponsorships available: Prospectus

© 2019 Cloud Native Computing Foundation

Who am I

- Well known as the “Cyber Guardian”
- Cybersecurity Strategist at Wallarm
- An Award-winning Cybersecurity Professional
- Founder and CEO of XR Safety Initiative
- Former Information Security Director Linden Lab
- Former Facebook Third Party Security Risk Advisor

Personal interests :

Emerging Technologies, Gaming, Virtual Worlds

Who am I - Ty Sbano

Ty Sbano is an Information Security Practitioner with 13 years of experience, mainly in Financial Technology organizations. Currently serving as Cloud Chief Information Security Officer at Sisense and advising for Watchertower.ai and Spherical Data. Previously Ty focused on leading application, product, and information security programs at >

Education

Penn State University

B.S. Information Science and Technology

Norwich University

M.S. Information Assurance

Certifications

CISSP, SSCP, CEH, CCSK, CPT

SISENSE

Periscope Data by Sisense

LendingClub

TARGET

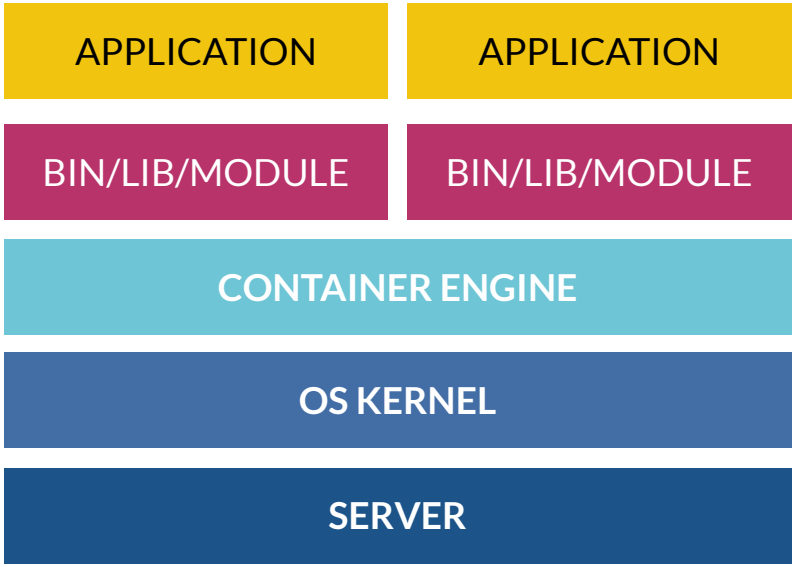
CapitalOne

JPMorganChase

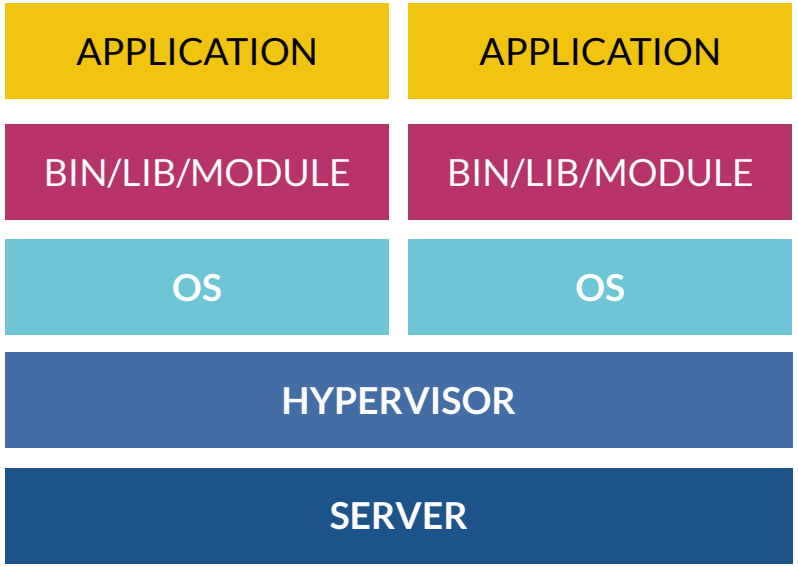
protiviti

Face the Future with Confidence

Overview of Containers

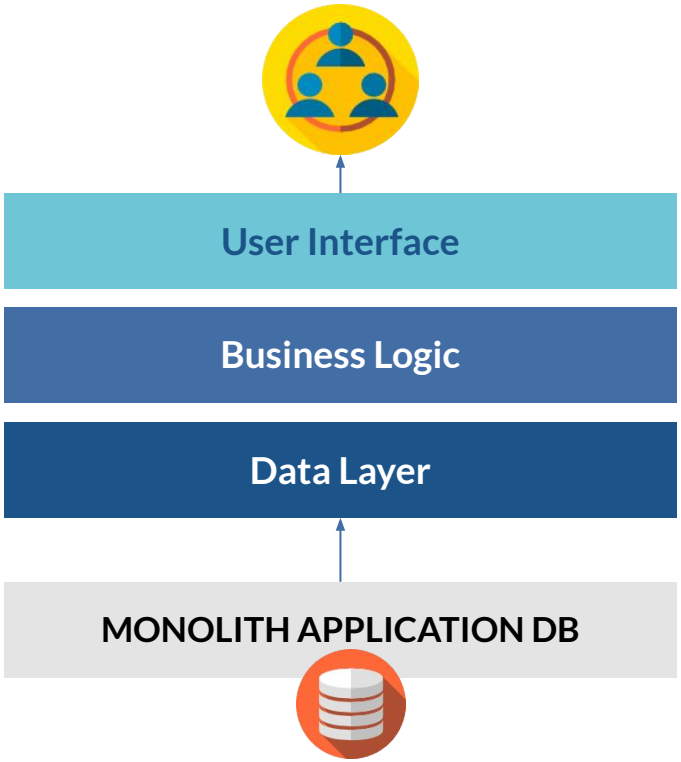


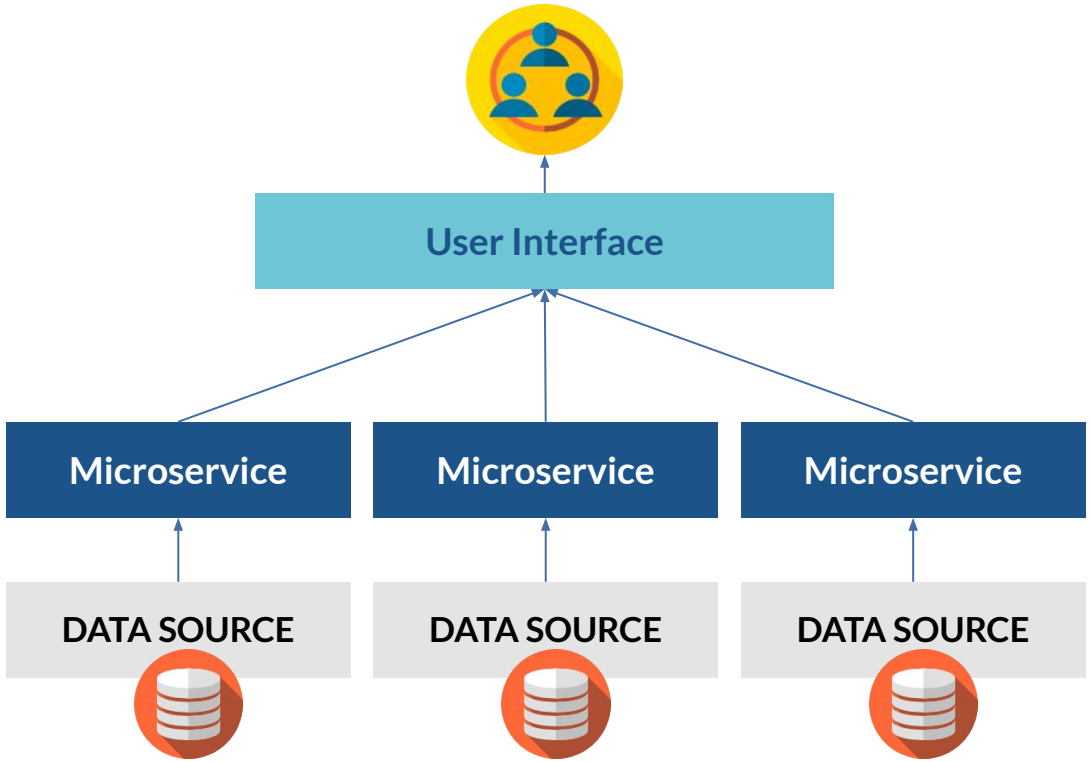
CONTAINER



VIRTUAL MACHINE

Monolith vs. Microservices?





What is up with these directions?

- North-South
  - Container to Clients
- East-West
  - Between Clusters/Pods

We hear a lot about Kubernetes, Kube, K8s...

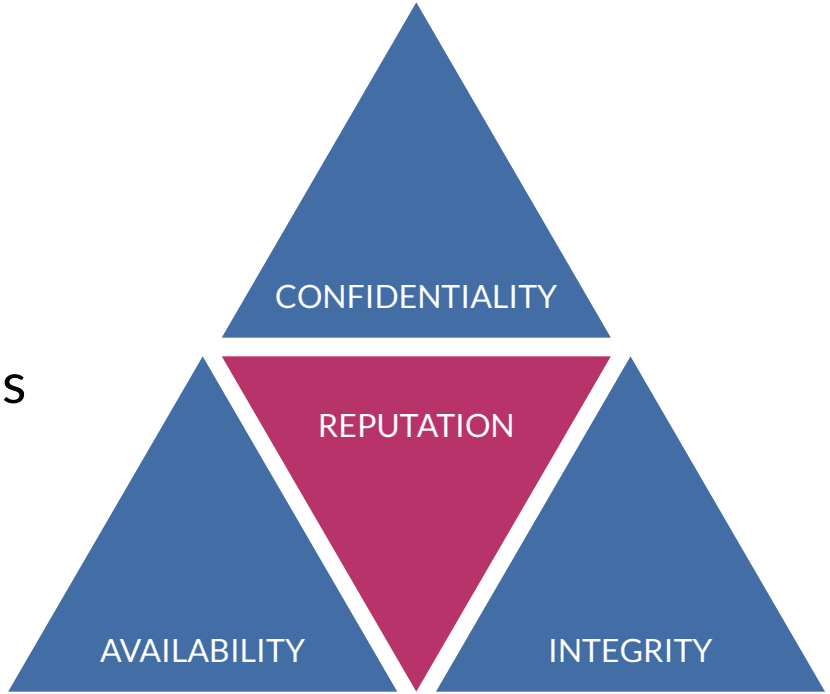
"koo-burr-NET-eez"

OPEN SOURCE CONTAINER ORCHESTRATION ENGINE

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. It has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available. The name Kubernetes originates from Greek, meaning helmsman or pilot.

Where to start?

- Inventory
- Leverage **Service Mesh**
- Risk Rank Process - Update Process
  - CIA Triad



Secure Defaults - Natively

- **NameSpace** - *A way to divide resources*
  - https://kubernetes.io/docs/concepts/overview/working-with-objects/namespaces/
- **Network Policy** - *Ingress & egress rules*
  - https://kubernetes.io/docs/concepts/services-networking/network-policies/
- **Pod Security Policy** - *Min Reqs to be accepted*
  - https://kubernetes.io/docs/concepts/policy/pod-security-policy/
- **Security Context** - *Applies to all containers in the Pod*
  - https://kubernetes.io/docs/tasks/configure-pod-container/security-context/

CLOUD NATIVE COMPUTING FOUNDATION

Secure Defaults - Natively

- **AppArmor** - *Protect and reduce attack surface*
  - https://kubernetes.io/docs/tutorials/clusters/apparmor/
- **Disabling Default Services** - *Least Privilege*
  - https://cloud.google.com/kubernetes-engine/docs/how-to/hardening-your-cluster
- **Certificate Management** - *Self-signed w/ kubeadm*
  - https://kubernetes.io/docs/tasks/administer-cluster/kubeadm/kubeadm-certs/
- **Back-ups** - *Encrypt them!*
  - https://kubernetes.io/docs/tasks/administer-cluster/encrypt-data/

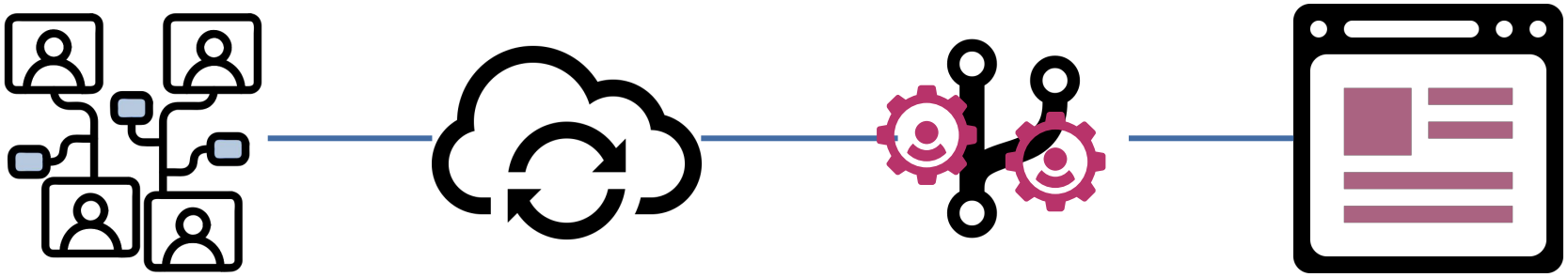
CLOUD NATIVE COMPUTING FOUNDATION

Production Access

- Who NEEDS access, why?
- Leverage Role Based Access Controls (RBAC)
  - https://kubernetes.io/docs/reference/access-authn-authz/rbac/
- Establish Risk Based Alerts
  - Event / Action
  - Time

CLOUD NATIVE COMPUTING FOUNDATION

Software Supply Chain



SUPPLIERS | WAREHOUSES | DEVELOPERS | PRODUCT

Security EcoSystem

- Where are your containers from?
  - Trusted Images (latest?)
- Security Scanning (Open Source)
  - Clair - https://coreos.com/clair/docs/latest/
  - Klar - https://github.com/optiopay/klar

Summary

- Security Hygiene FTW
  - Inventory
  - Hardening
  - Scanning

kubernetes

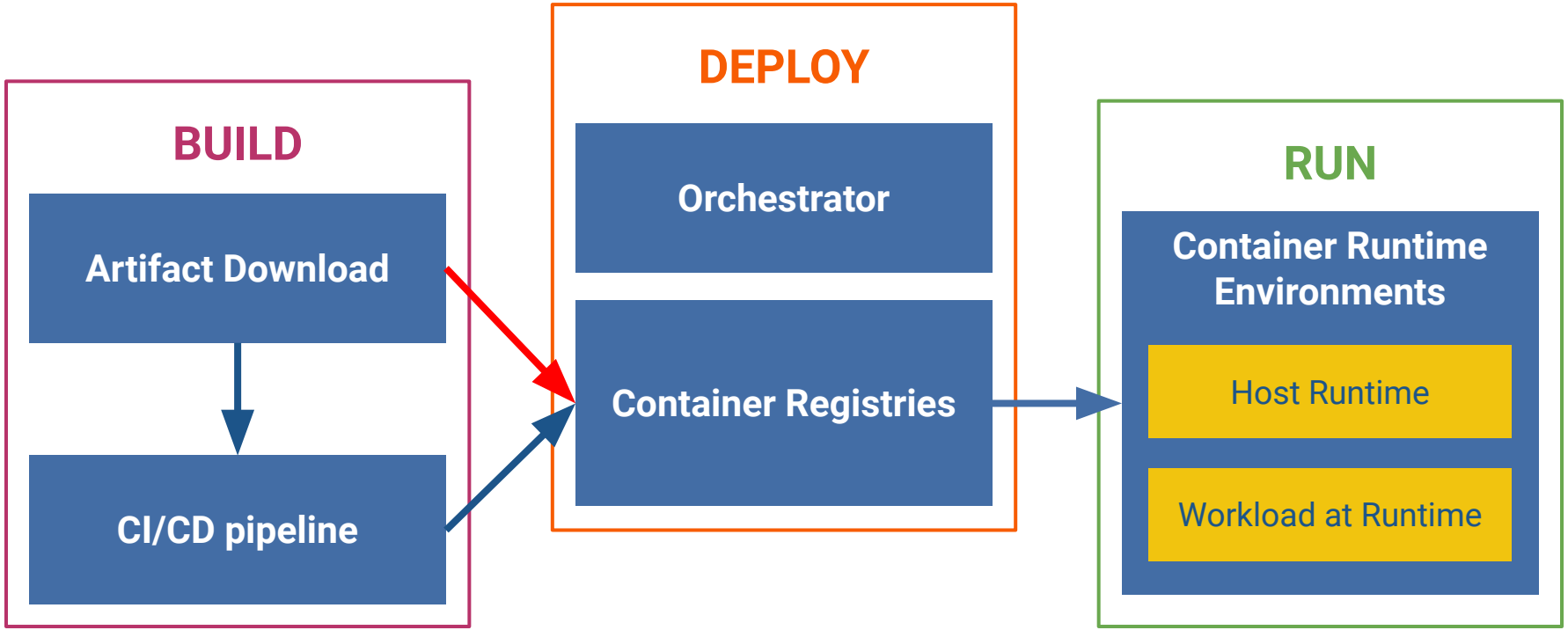
DOs and DON'Ts
Container Security

Getting Started

"Cloud-native applications and infrastructure create several new challenges for all of us security professionals. We need to establish new security programs, have a new mindset and adopt advanced new tools that are focused primarily on securing cloud-native technologies."

- Kavya Pearlman

Build. Deploy. Run.



Build. Deploy. Run.

BUILD TIME CONSIDERATIONS

- Application Security
  - Secure Coding Practices
  - SAST/DAST
- Image Scanning on Build/Pull
  - Vulnerability Management
  - SCA - Software Composition Analysis
- Image Signing
- Attack Surface Reduction
  - Multi-Stage Builds

Build. Deploy. Run.

DEPLOY TIME CONSIDERATIONS

- Image Registries
- Vulnerability Management
  - Regular Scans
  - Maintain deployment Info
- RBAC - Limit User Privileges
- Configuration Manager
  - Open APIs
- Secrets Management Integration
- Traffic Segregation

Build. Deploy. Run.

RUN TIME CONSIDERATIONS

- Host Protection
- Hardening
  - CIS Benchmarks
  - Container-Friendly Hosts
- Network Segregation
  - Protect APIs
- Container Firewalls
- Activity monitoring, logging & auditing
- Patching & Vulnerability Tracking

Container Hardening

Center for Internet Security® **COMES TO THE RESCUE!!**

- Downgrade to non privileged user

```
RUN adduser -D limited_user
USER limited_user
```

- OR provide the following option at runtime:

```
docker run -u limited_user ubuntu
```

- Mitigate Denial of Service by limiting resource usage

```
docker run -it --cpus 1 --memory 512Mb ubuntu
```

- Enforce a good AppArmor Profile

```
https://github.com/genuinetools/bane
```

Host Protection

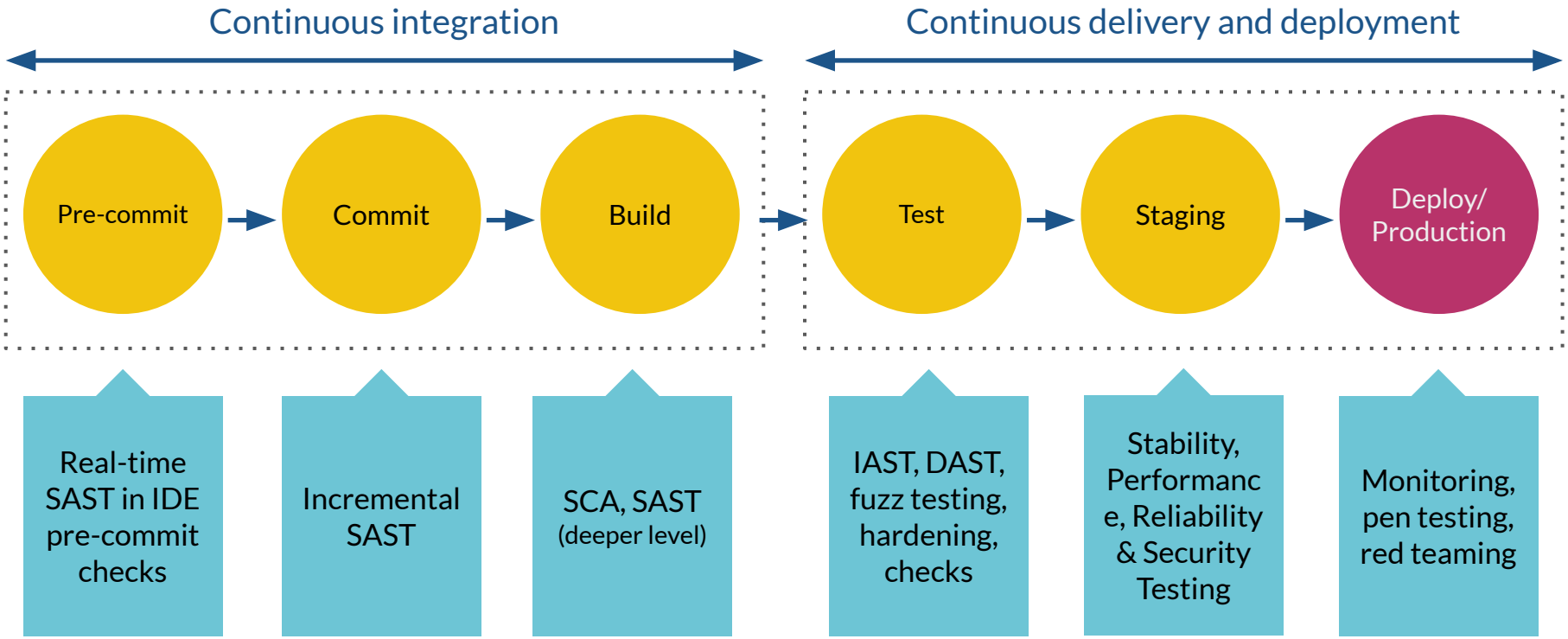
Lock Down the Host (Volume write / exec, etc.)

Use Seccomp to to restrict Host syscall access

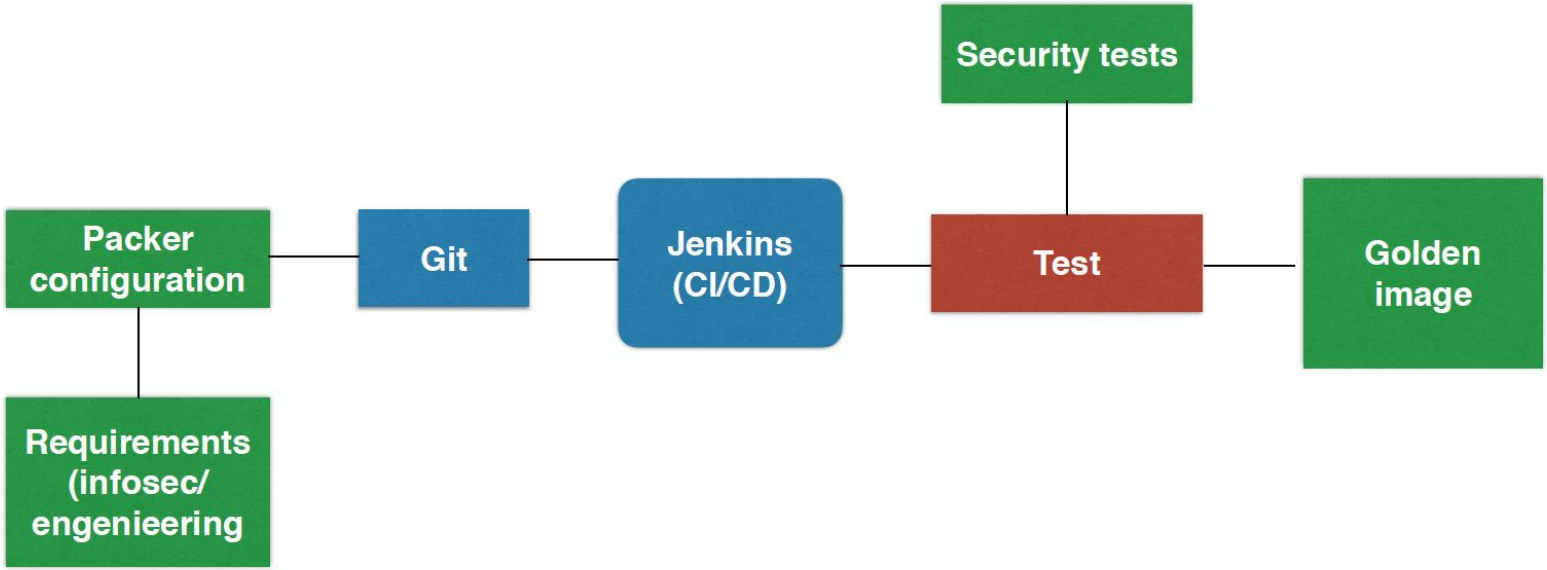
SELinux to prevent container escape

Review NIST 800-190 for detailed guidelines

Security Tooling



Infrastructure as Code



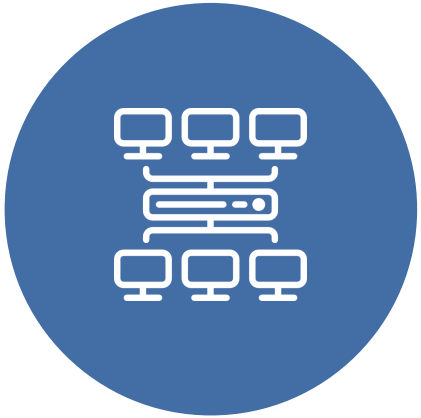
Open Source Tools For Container Security

LINKERD

DOs for Containerized Environments

CREATE IMMUTABLE CONTAINERS

RUN IMAGES ONLY FROM TRUSTED SOURCES



USE CONTAINER-NATIVE MONITORING TOOLS

NOT To Dos for Containerized Environments

- Installing an operating system inside a Docker container
- Running unnecessary services
- Storing critical data inside a container
- Hard-Coded Credentials for accessing Registry
- Hosting too many services inside a container

Q&A

Kavya and Ty Contact Info

Linkedin **kavya-pearlman**

Twitter **@KavyaPearlman**

Website - **www.wallarm.com**

Linkedin **tysbano**

Twitter - **@tysbano**

Website - **tysbano.com**

Thank You!