Securing the Perimeter
CFCR/CFAR Chain Of Custody With CI/CD Pipelines
Who We Are...

Keith Strini ...

- Provide operational assistance and guidance
- Build balanced customer product teams delivering “Platform” as a capability within their organization
- Establish and maintain continuous delivery pipelines for deployment of Pivotal Cloud Foundry and related products in a customer’s infrastructure
- Design and implement continuous integration and continuous delivery processes to deliver customer applications to production, fostering a culture of continuous process improvement

.... Shaozhen Ding
Shift the Mindset

"Assume The Continuous Threat of Compromise and Then Continuously Move The Target"

From Reactive Courses of Action into Proactive Security Policy

- **Recovery Point Objective**
  - The recovery point objective (RPO) is the point in time that you wish to recover to.

- **Recovery Time Objective**
  - The recovery time objective (RTO) is how long it takes to recover, taken irrespective of the RPO. That is, after the disaster, how long until you have recovered to the point determined by the RPO.
A Standard Delivery Pipeline

DVCS → Build → Artifactory

- Deploy To Test
- Deploy To QA
- Deploy To Prod

binary/containers

Platform As A service
Defending the Supply Chain Threat
Defending the Application Integrity

- DVCS
- Build
- Artifactory
- Deploy To Test
- Deploy To QA
- Deploy To Prod
- Platform As A Service

Commit Hash → Hash Verification → Binary Hash → Build → Artifactory → Hash Verification → Deploy To Test, QA, Prod
Enterprise Docker Registry - Harbor

- LDAP Integration
- Clair Security Scanning
- Docker Content Trust via Notary
- Role Based Access Control
Defending the Continuous Threat
(Privilege Escalation, File System Modification, Container Breakout)

User Namespace => Restricted User Scope

Mount Namespace + pivot_root => File System Isolation

Each Container can have its own root filesystem separate from the host

- PCF uses hardened, streamlined Ubuntu stemcell
- VMs use hardened, streamlined cflinuxfs3 rootFS
- PCF uses a combination of OverlayFS and XFS as a filesystem for containers
- The read-only layer in all containers is RootFS
- The application binaries are in a very small read-write layer of the file system
Defending the Continuous Threat
(DoS of Containers, DoS of Service to Host, Kernel Mods, MITM Attacks)

Network Namespace + port virtualization => Network Isolation
CGroups => Resource Isolation (CPU share Capping), Device Access WL
Rootless containers
App Armor confines untrusted processes
Seccomp system call filtering
Defending the Continuous Threat
(Malware)

Continuous Zero Downtime CVE Patching
(Repair Vulnerable Components)
Defending the Continuous Threat
(Advanced Persistent Threat)

Canary Style deployment model plus Infrastructure as code from version control (Repave all system components)
Defending the Continuous Threat
(Leaked Credentials)

Continuous Zero
Zero Trust Network
Model (Rotation of all
system credentials)
BOSH is an open source tool for release engineering, deployment, lifecycle management, and monitoring of distributed systems such as Kubernetes.

- Packaging w/ embedded OS
- Server provisioning on any IaaS
- Software deployment across availability zones
- Health monitoring (server AND processes)
- Self-healing w/ Resurrector
- Storage management
- Rolling upgrades via canaries
- Easy scaling of clusters
- Backup and Restore
- Rotating Server Credentials

Cloud Native Security
- Repair
- Repave
- Rotate
Bringing it all together

Application Code & Frameworks
- Buildpacks
- Spring Boot
- Spring Cloud
- Steeltoe

YOU build the container

Pivotal Application Service (PAS)
> cf push

Java | .NET | NodeJS

CPI (15 methods)

Embedded OS
(Windows & Linux)

YOU build the container

Pivotal Container Service (PKS)
> kubectl run

Elastic | Packaged Software | Spark

Open Service Broker API

Pivotal Services Marketplace
- Pivotal and Partner Products

Public Cloud Services

Customer Managed Services

WE build the container

GitHub

Concourse

CPEs
Product Updates

Continuous delivery

Openstack

Pivotal Network

Concourse

vSphere

Azure & Azure Stack

Google Cloud

AWS

Elastic

Spark

v1

v2

v3

...
Conclusion - Move The Target

Proactive Security Policy

Phase I - Aggressive Rotation of the issued Developer Keys

Phase II – Rotation of Environment Credentials, End Point IPs, and Dynamic Management of IP WL/ACLs

Phase III - Continuous Verification of the Application Integrity

Phase IV – Continuous Authorization for Runtime Validation

Phase V – Continuous Paving of the Environment, Rotation of the Keys, Renewal of Authorizing Credentials plus Least Privilege Container Authority