Red Hat IRS Mini Summit 2 (OpenShift Security)

9/18 NCFB

Mark Hilburger, Account Exec 703-217-4511 mhilburg@redhat.com

Eamon McCormick, Emerging Technologies, 443-413-2719 emccormi@redhat.com

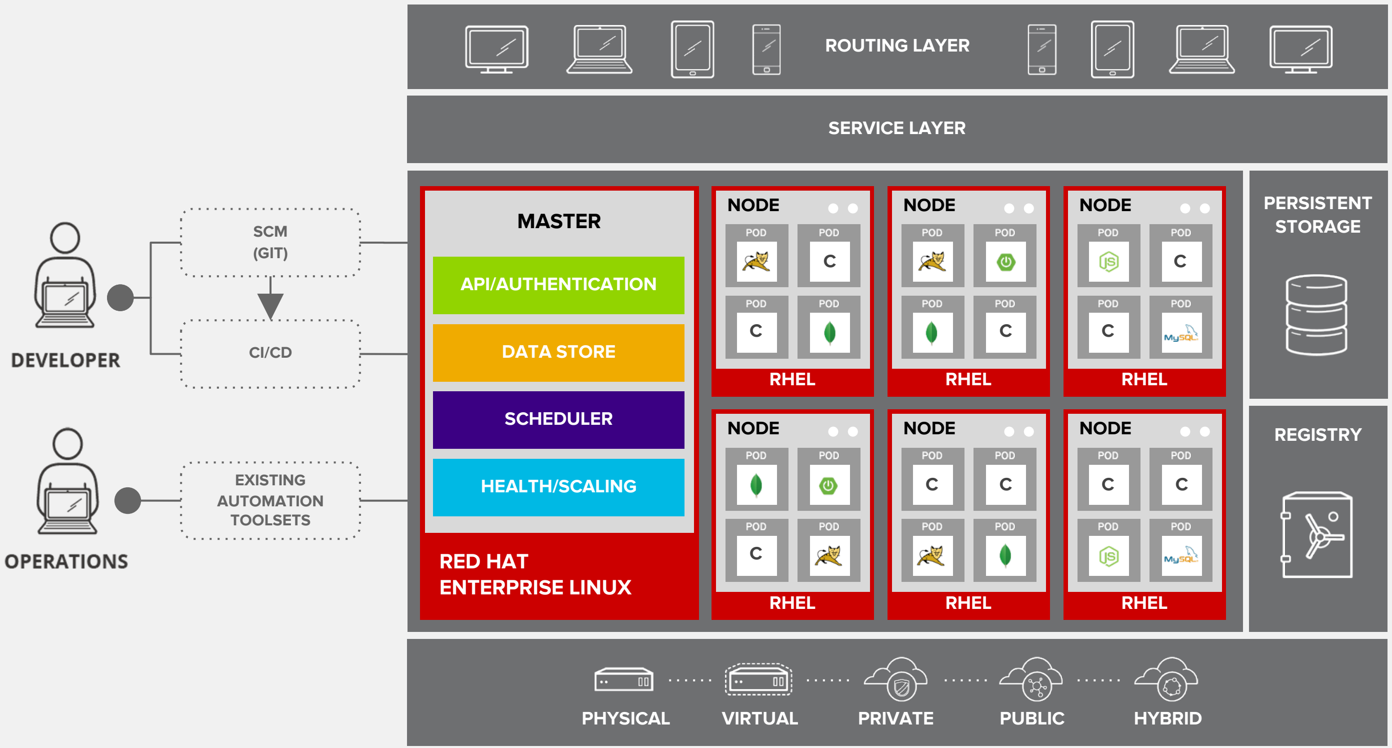
AGENDA

8:00 AM	Red Hat overview and team intro	Mark Hilburger
8:10 AM	OpenShift Architecture and install process	Brandon Cox/ Eamon McCormick
9:00 AM	IRS Vision	Sharon James
9:40 AM	OpenShift Demo	Patrick Cunning
10:40 AM	**Break**	
10:55 AM	RHEL and container Security overview	Calvin Smith
11:25 AM	OSCAP Components	Shawn Wells
11:55 AM	**lunch**	
12:55 PM	OpenShift S2I for Hardened EAP	Khary Mendez
1:25 PM	CI/CD in OpenShift	Khary Mendez
1:55 PM	Atomic Scan in Jenkins pipeline	Khary Mendez
2:25 PM	**Break**	
2:40 PM	Scheduling OSCAP Scans for RHEL hosts	Cameron Wyatt
3:25 PM	OpenControl to automate Security Authorization Package	Shawn Wells
4:40 PM	**end**	

OpenShift Install and Overview

Eamon McCormick

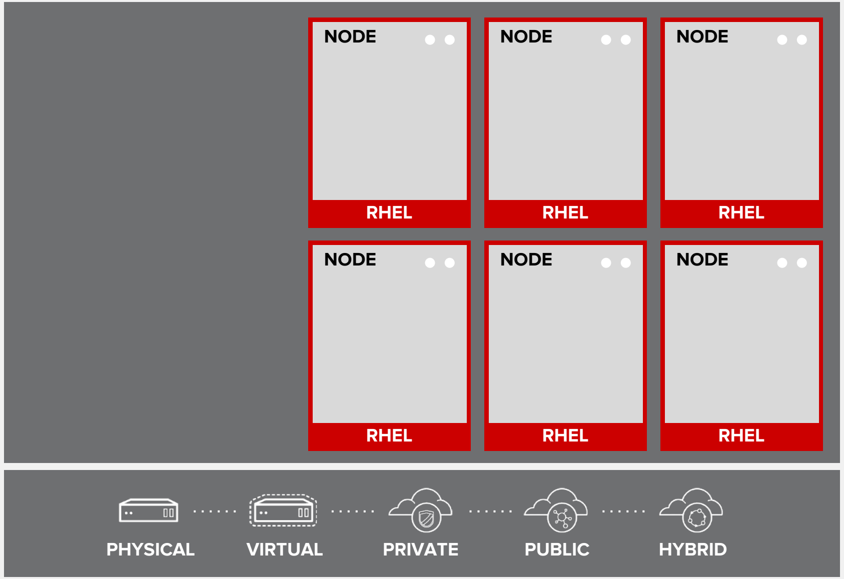
OPENSHIFT ARCHITECTURE



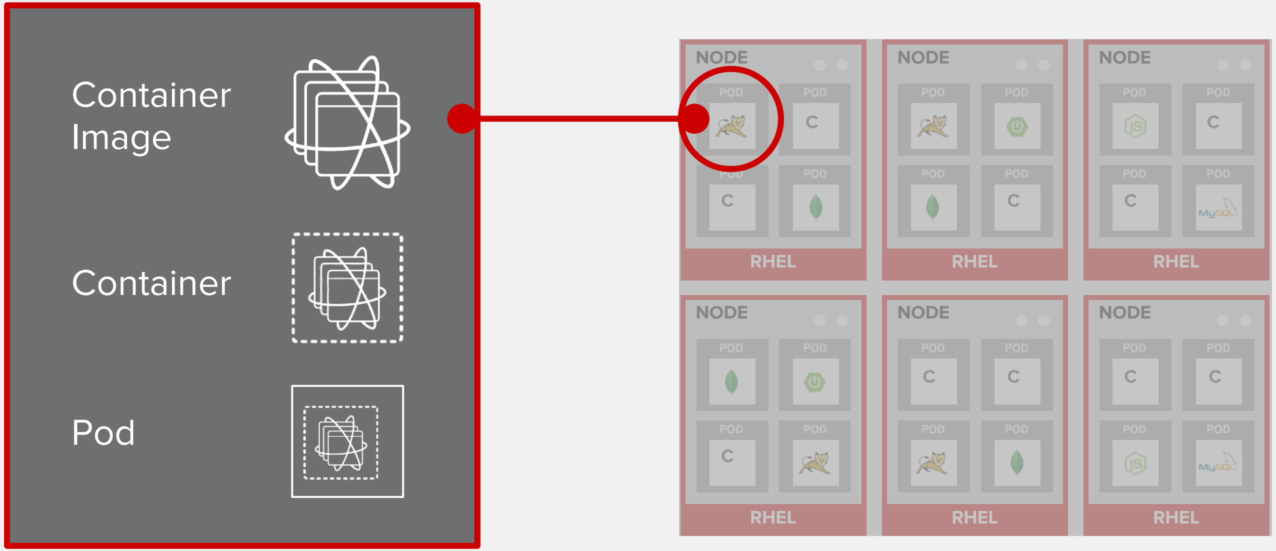
YOUR CHOICE OF INFRASTRUCTURE

PHYSICAL · VIRTUAL · PRIVATE · PUBLIC · HYBRID

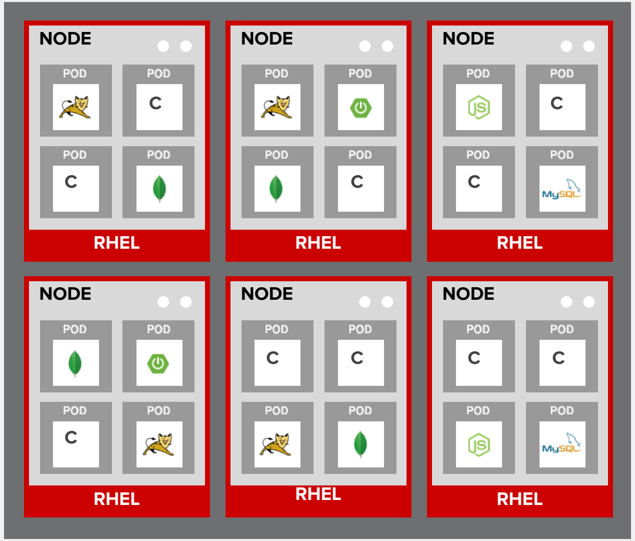
NODES RHEL INSTANCES WHERE APPS RUN



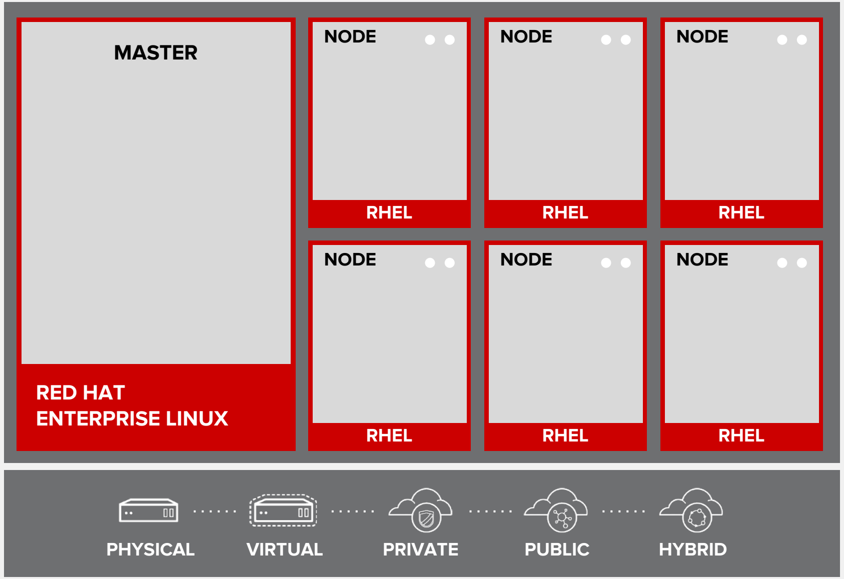
APPS RUN IN CONTAINERS



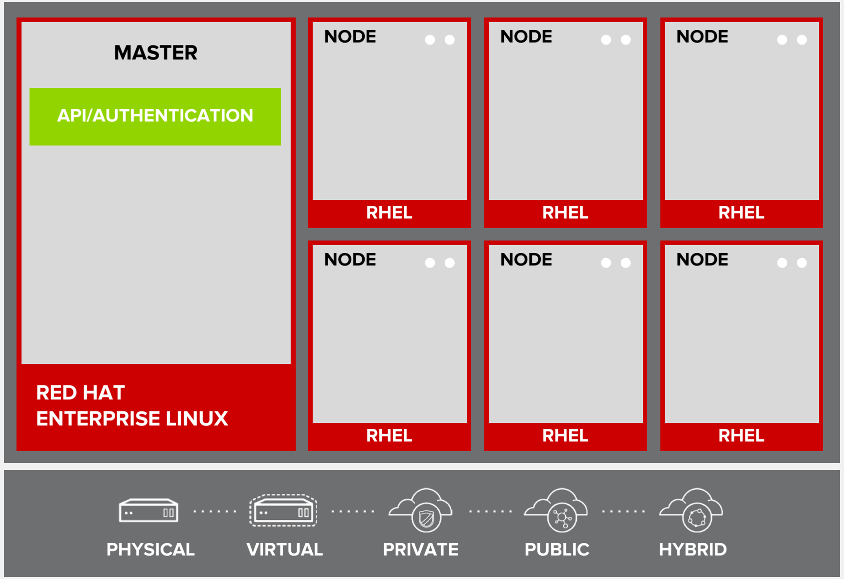
PODS ARE THE UNIT OF ORCHESTRATION



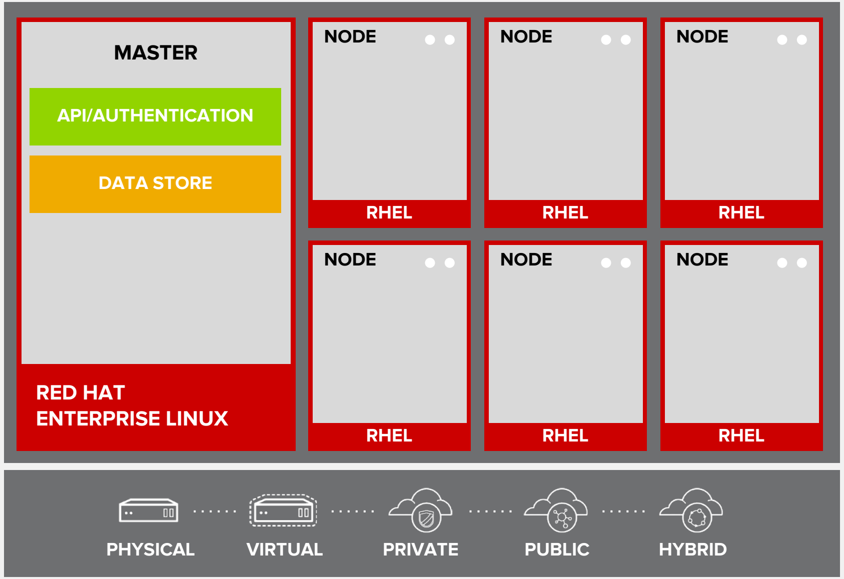
MASTERS ARE THE CONTROL PLANE



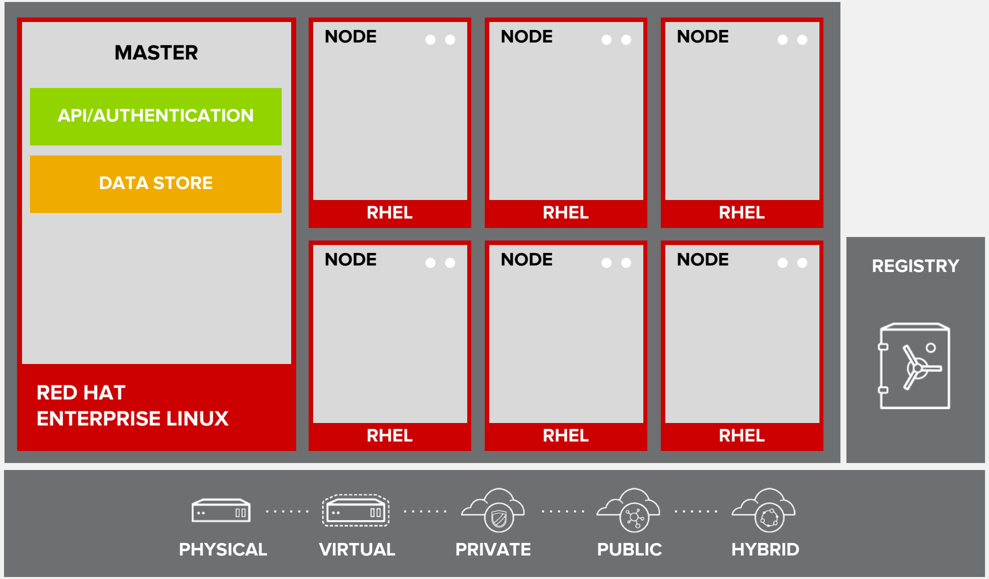
API AND AUTHENTICATION



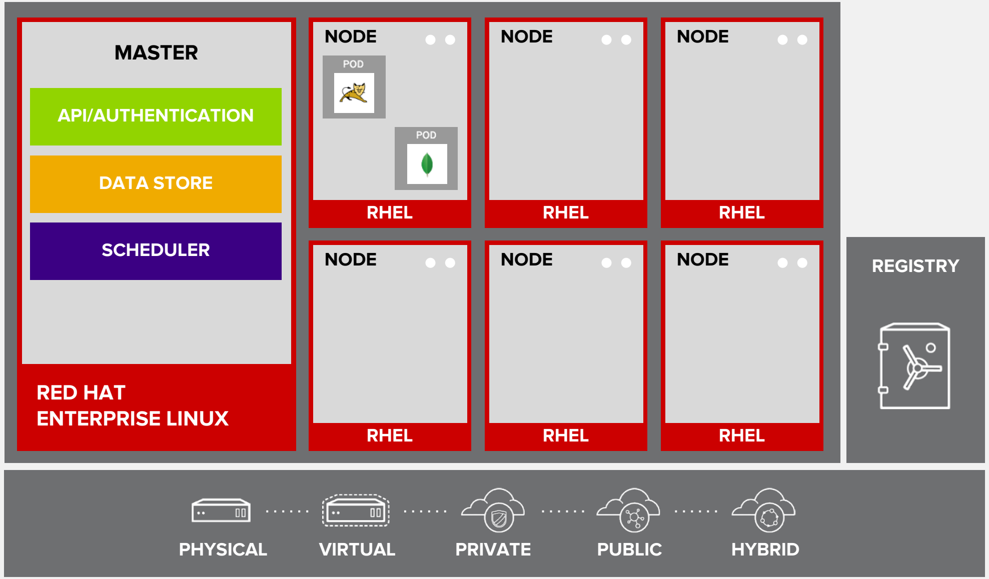
DESIRED AND CURRENT STATE



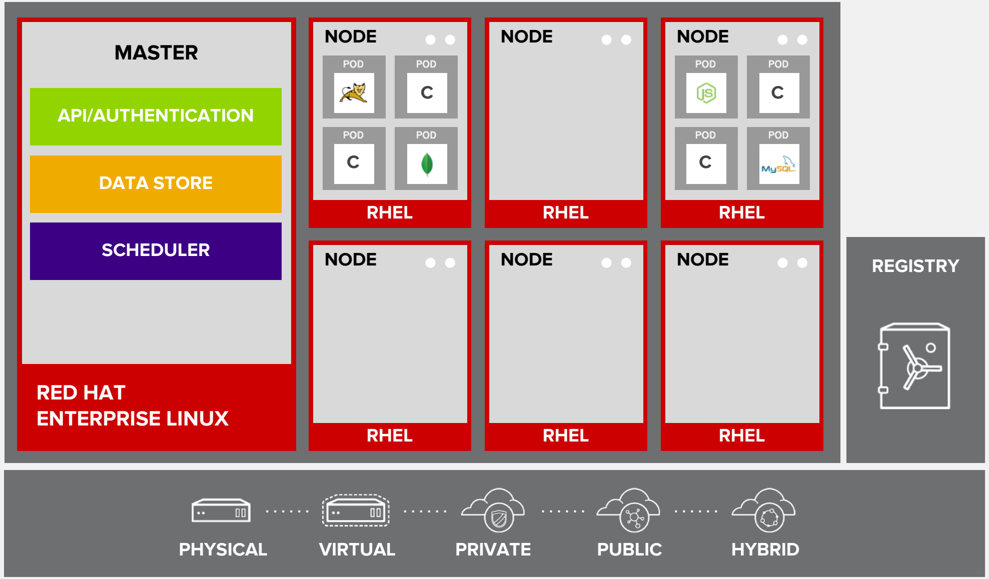
INTEGRATED CONTAINER REGISTRY



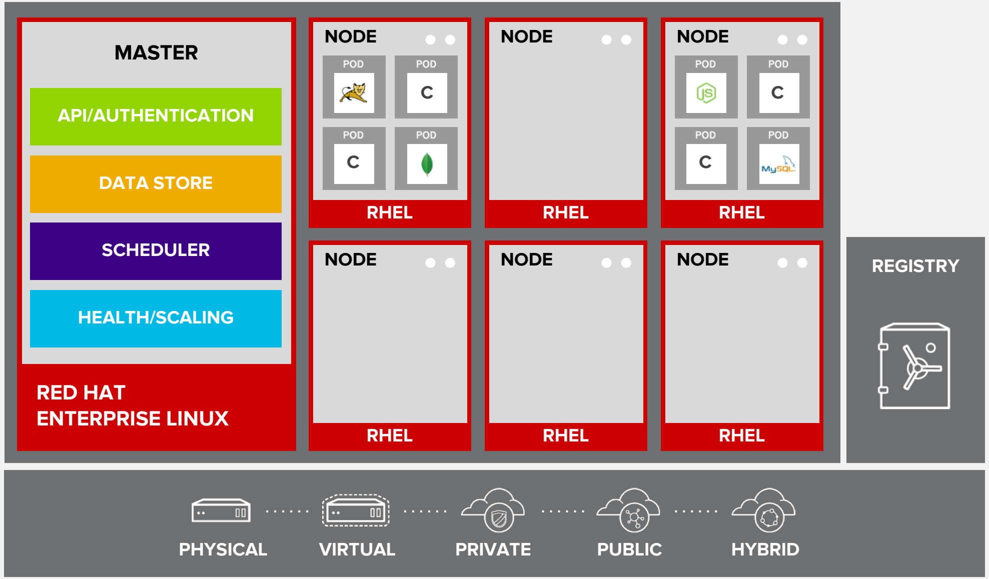
ORCHESTRATION AND SCHEDULING



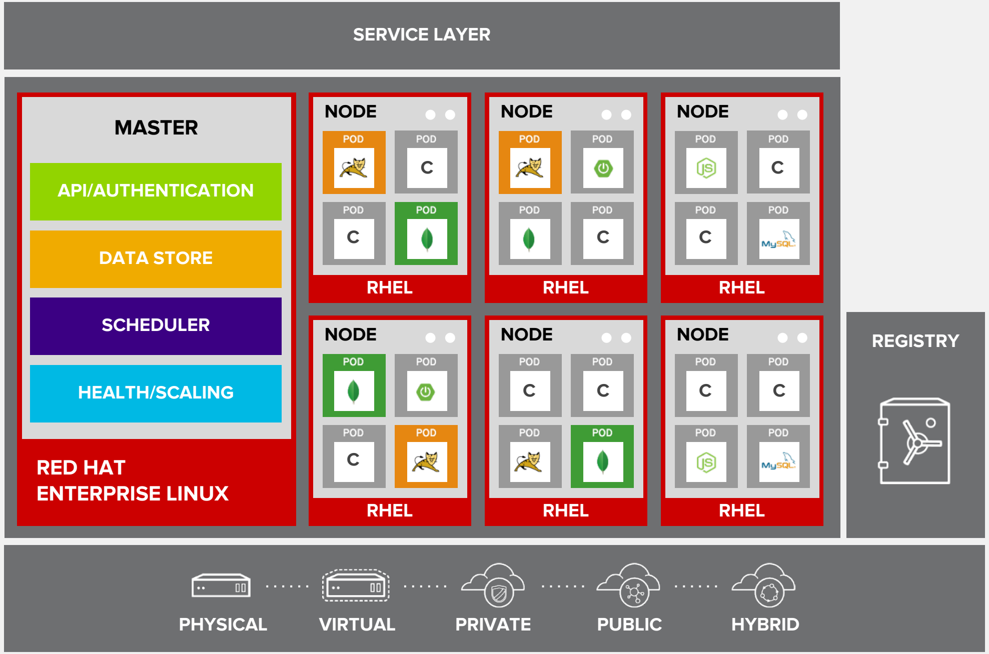
PLACEMENT BY POLICY



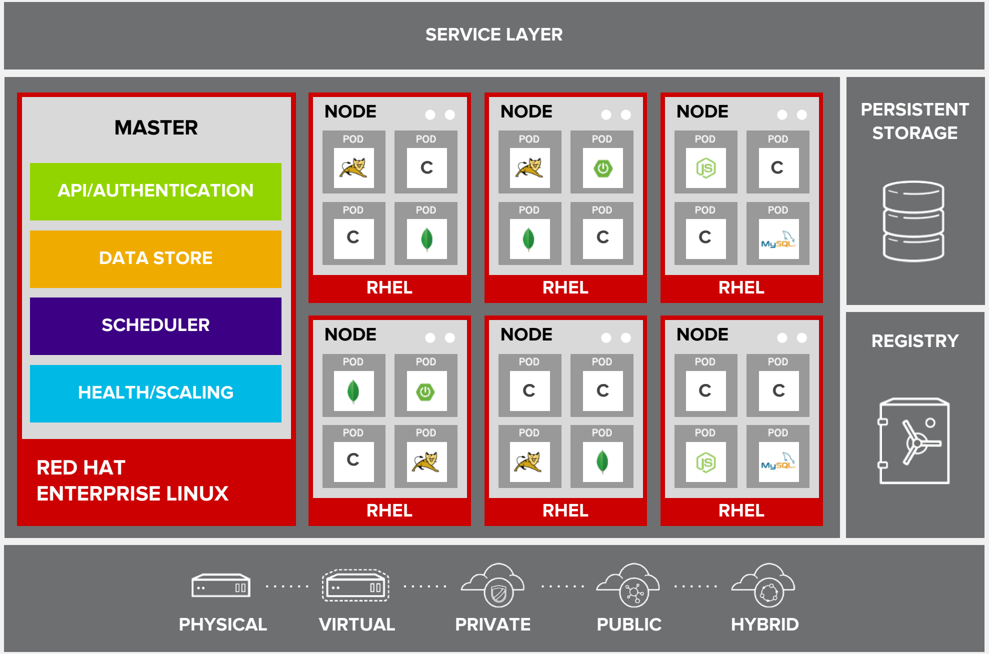
AUTOSCALING PODS



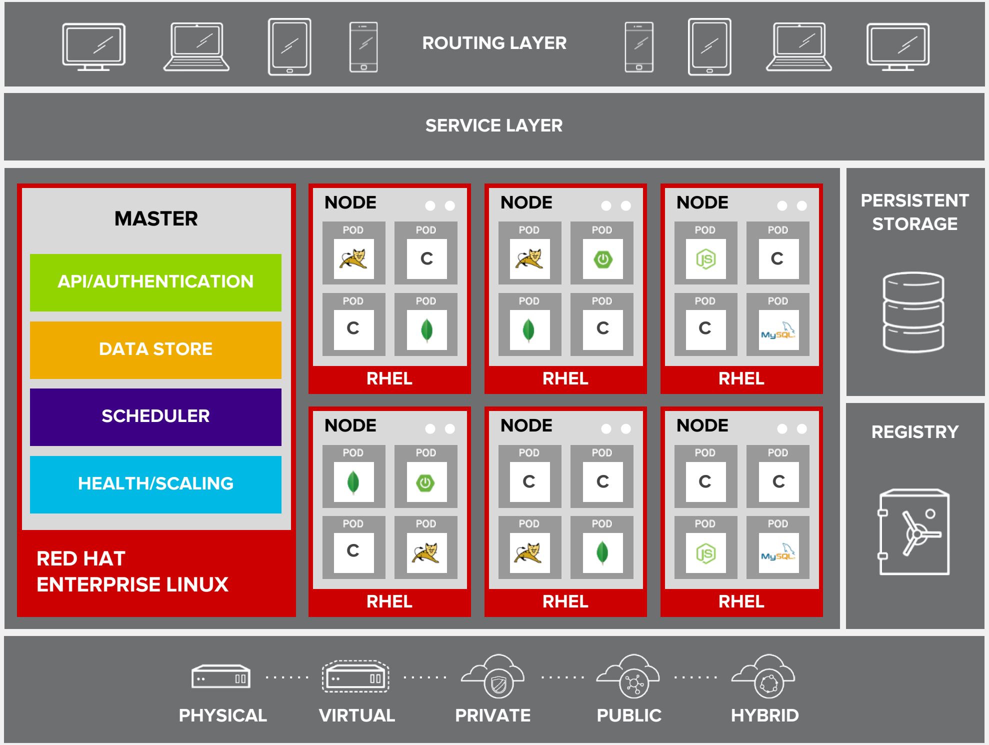
SERVICE DISCOVERY



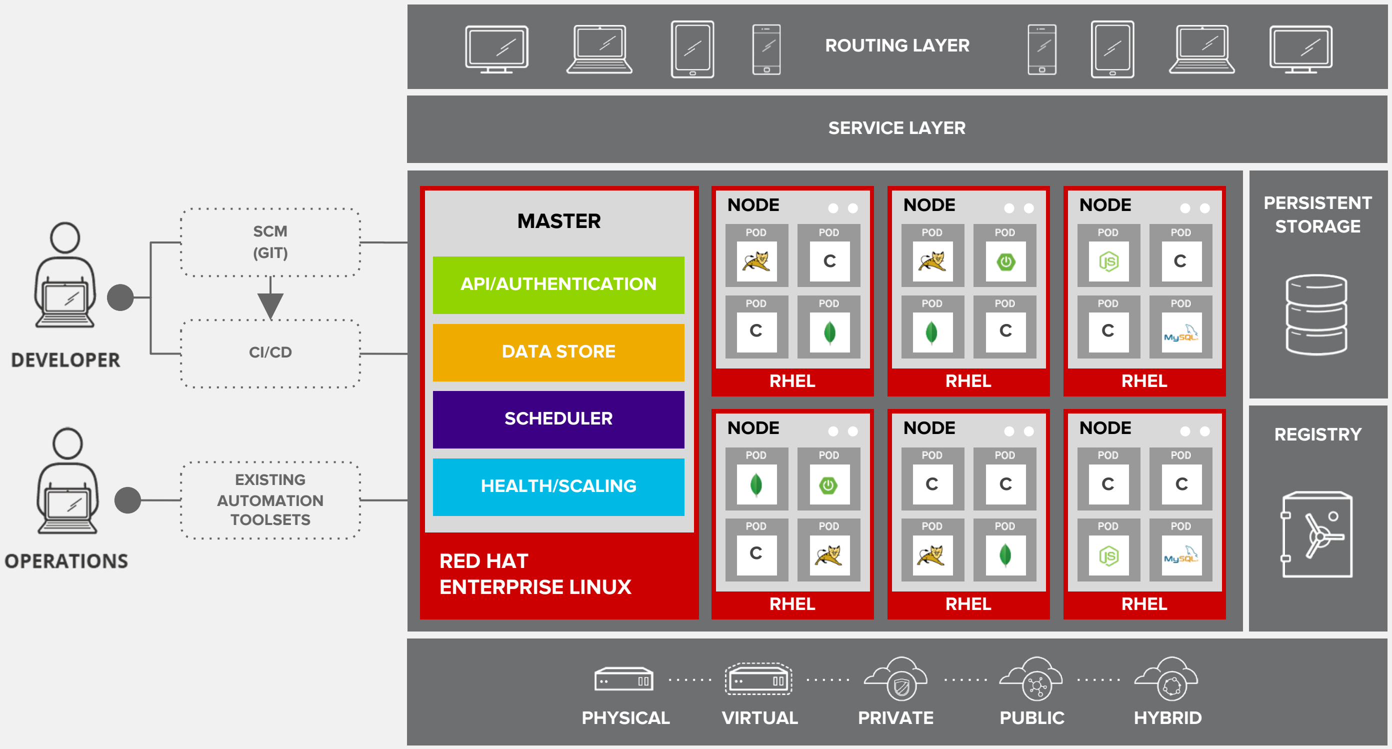
PERSISTENT DATA IN CONTAINERS



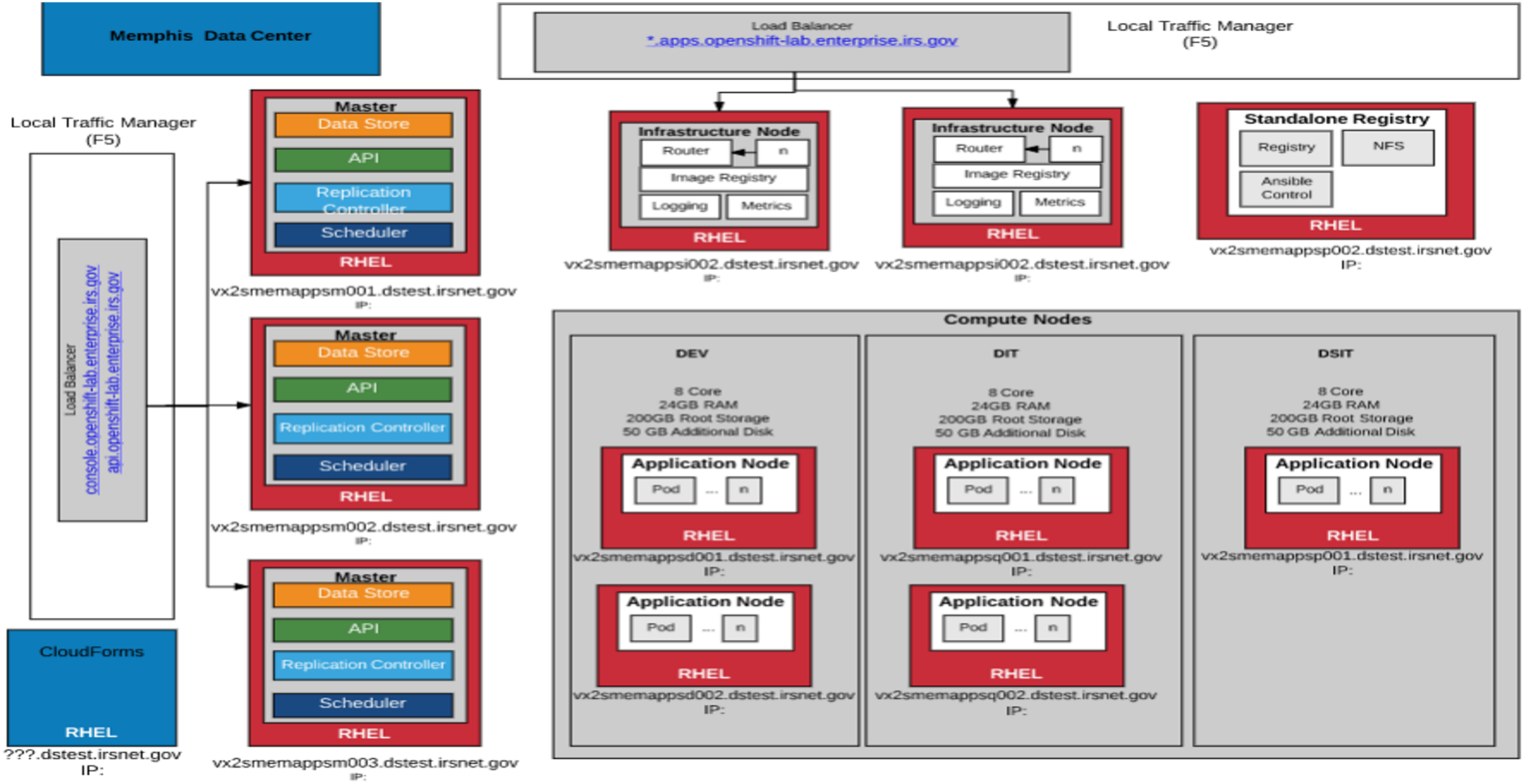
ROUTING AND LOAD-BALANCING



ACCESS VIA WEB, CLI, IDE AND API



The IRS Deployment



Memphis Data Center

Local Traffic Manager
(F5)

Load Balancer
console.openshift-lab.enterprise.irs.gov
api.openshift-lab.enterprise.irs.gov

CloudForms
RHEL
???.dstest.irsnet.gov
IP:

Master
Data Store
API
Replication Controller
Scheduler
RHEL
vx2smemappsm001.dstest.irsnet.gov
IP:

Master
Data Store
API
Replication Controller
Scheduler
RHEL
vx2smemappsm002.dstest.irsnet.gov
IP:

Master
Data Store
API
Replication Controller
Scheduler
RHEL
vx2smemappsm003.dstest.irsnet.gov
IP:

Load Balancer
*.apps.openshift-lab.enterprise.irs.gov

Local Traffic Manager
(F5)

Infrastructure Node
Router
n
Image Registry
Logging
Metrics
RHEL
vx2smemappsi002.dstest.irsnet.gov
IP:

Infrastructure Node
Router
n
Image Registry
Logging
Metrics
RHEL
vx2smemappsi002.dstest.irsnet.gov
IP:

Standalone Registry
Registry
NFS
Ansible Control
RHEL
vx2smemappsp002.dstest.irsnet.gov
IP:

Compute Nodes

DEV
8 Core
24GB RAM
200GB Root Storage
50 GB Additional Disk

Application Node
Pod ... n
RHEL
vx2smemappsd001.dstest.irsnet.gov
IP:

Application Node
Pod ... n
RHEL
vx2smemappsd002.dstest.irsnet.gov
IP:

DIT
8 Core
24GB RAM
200GB Root Storage
50 GB Additional Disk

Application Node
Pod ... n
RHEL
vx2smemappsq001.dstest.irsnet.gov
IP:

Application Node
Pod ... n
RHEL
vx2smemappsq002.dstest.irsnet.gov
IP:

DSIT
8 Core
24GB RAM
200GB Root Storage
50 GB Additional Disk

Application Node
Pod ... n
RHEL
vx2smemappsp001.dstest.irsnet.gov
IP:

INSTALLATION

Installation Process - Ansible

- What is Ansible?
- Ansible Layout of the Advanced installer
  - /etc/ansible/hosts
  - /usr/share/ansible/openshift-ansible/playbooks/byo/config.yml
- Example execution of Ansible playbook for OCP

```
ansible-playbook  [-i /path/to/inventory] /usr/share/ansible/openshift-ansible/playbooks/byo/config.yml
```

Things to Consider

- Which installation method do you want to use?
- How many hosts do you require in the cluster?
- How many pods are required in your cluster?
- Is high availability required?
- Which installation type do you want to use: RPM or containerized?
- Is my installation supported if integrating with other technologies?

Installation Overview

- Installation Methods - Quick Installation vs Advanced Installation
- Sizing Considerations
- Environment Scenarios
  - Single Master - Multiple Nodes
  - Single Master - Multiple Nodes - Multiple etcd
  - Multiple Masters
- RPM vs Containerized

Upgrading A Cluster

- In-place (manual or automated)
  - With in-place upgrades, the cluster upgrade is performed on all hosts in a single, running cluster: first masters and then nodes. Pods are evacuated off of nodes and recreated on other running nodes before a node upgrade begins; this helps reduce downtime of user applications.

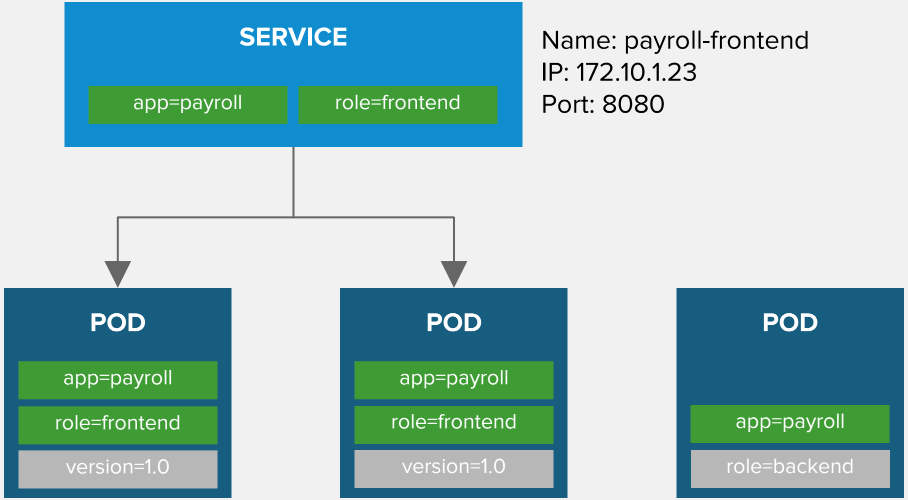
- Blue - Green Upgrades
  - masters and etcd servers are still upgraded first, however a parallel environment is created for new nodes instead of upgrading them in-place.

Process for Downgrading a Cluster

- Verify backups - etcd, config …
- Shut down the cluster
- Remove RPM's
- Downgrade docker (depends on version of OCP)
- Reinstall old RPM's
- Restore etcd
- Bring OCP services back on-line
- Verify that the downgrade was successful

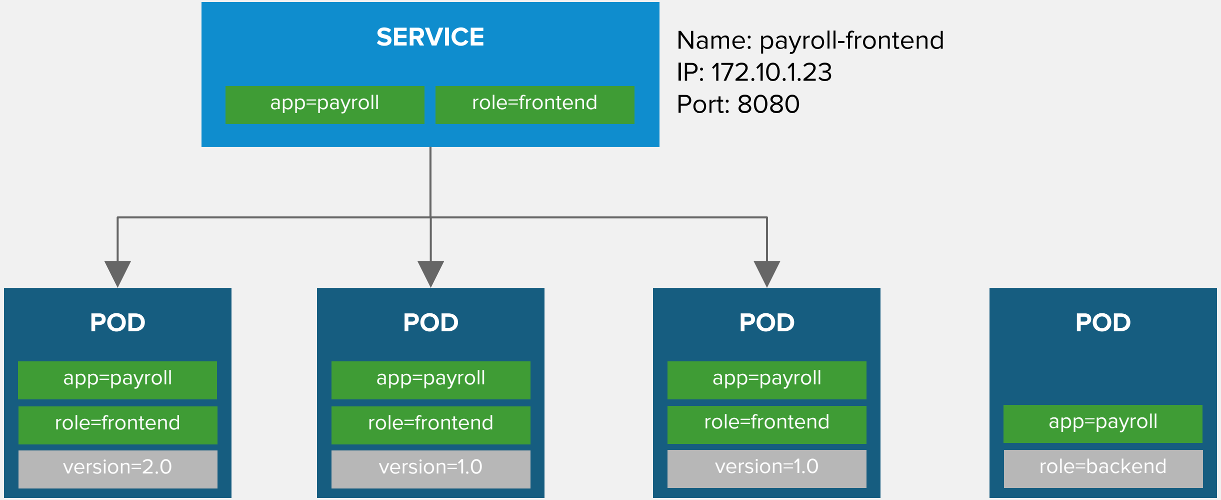
NETWORKING

BUILT-IN SERVICE DISCOVERY INTERNAL LOAD-BALANCING



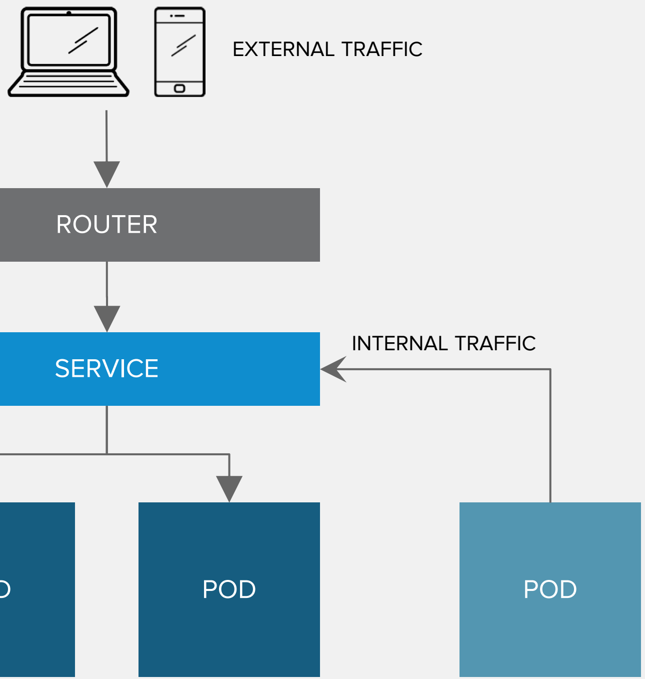
Name: payroll-frontend
IP: 172.10.1.23
Port: 8080

BUILT-IN SERVICE DISCOVERY INTERNAL LOAD-BALANCING



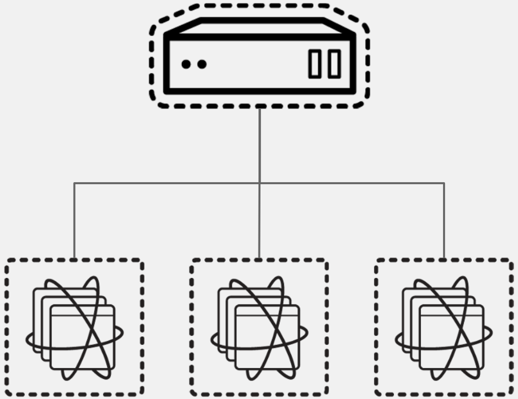
Name: payroll-frontend
IP: 172.10.1.23
Port: 8080

ROUTE EXPOSES SERVICES EXTERNALLY



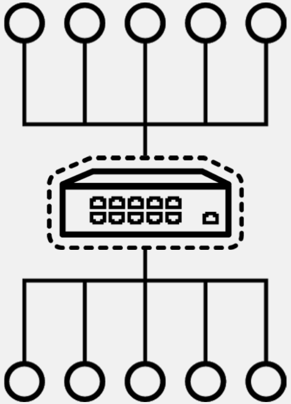
ROUTING AND EXTERNAL LOAD-BALANCING

- Pluggable routing architecture
    - HAProxy Router
    - F5 Router
- Multiple-routers with traffic sharding
- Router supported protocols
    - HTTP/HTTPS
    - WebSockets
    - TLS with SNI
- Non-standard ports via cloud load-balancers, external IP, and NodePort

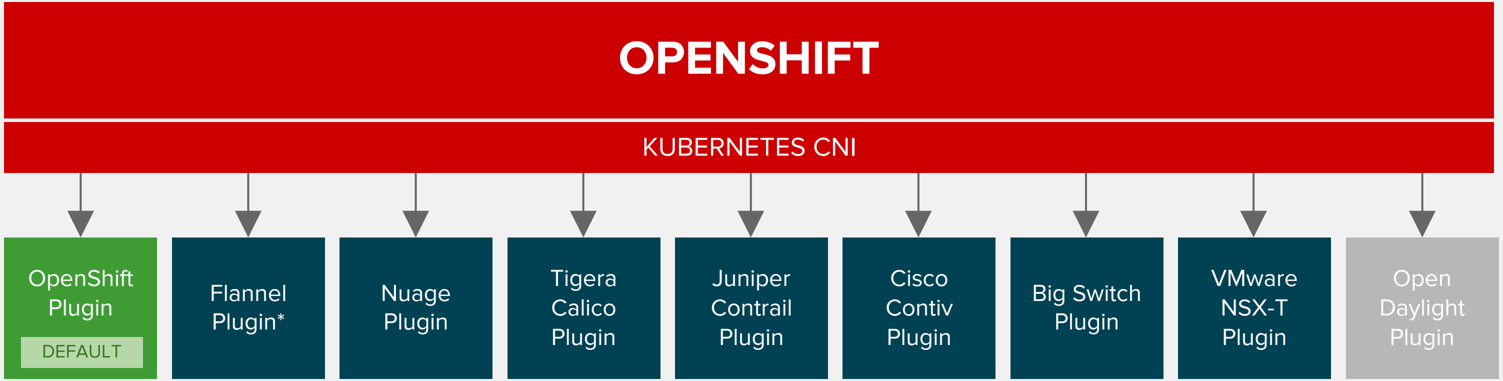


OPENSHIFT NETWORKING

- Built-in internal DNS to reach services by name
- Split DNS is supported via SkyDNS
  - Master answers DNS queries for internal services
  - Other nameservers serve the rest of the queries
- Software Defined Networking (SDN) for a unified cluster network to enable pod-to-pod communication
- OpenShift follows the Kubernetes Container Networking Interface (CNI) plug-in model



OPENSHIFT NETWORK PLUGINS



OPENSHIFT

KUBERNETES CNI

OpenShift Plugin DEFAULT

Flannel Plugin*

Nuage Plugin

Tigera Calico Plugin

Juniper Contrail Plugin

Cisco Contiv Plugin

Big Switch Plugin

VMware NSX-T Plugin

Open Daylight Plugin

🟩 Certified Plugin

■ Validated Plugin

▨ In-Progress

For a Complete List of Certified Plugins refer to OpenShift Third-Party SDN FAQ

* Flannel is minimally verified and is supported only and exactly as deployed in the OpenShift on OpenStack reference architecture

OPENSHIFT SDN

FLAT NETWORK (Default)

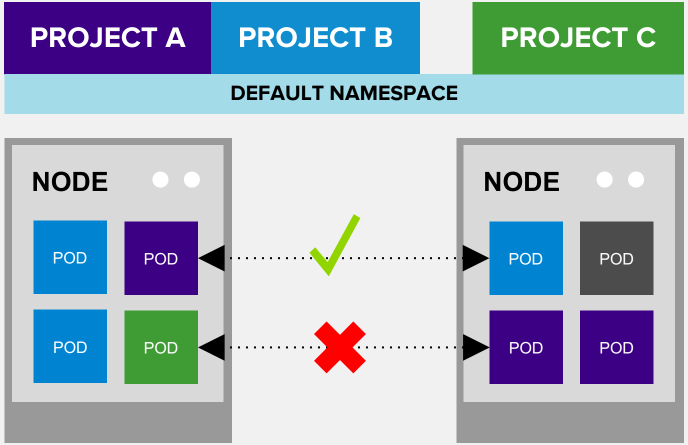
- All pods can communicate with each other across projects

MULTI-TENANT NETWORK

- Project-level network isolation
- Multicast support
- Egress network policies

NETWORK POLICY (Tech Preview)

- Granular policy-based isolation

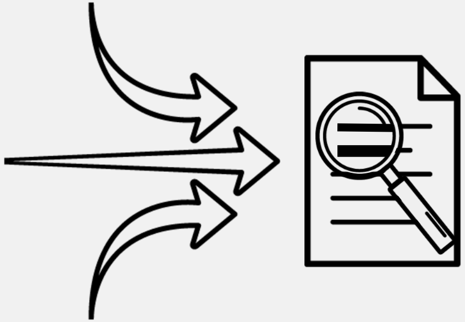


Multi-Tenant Network

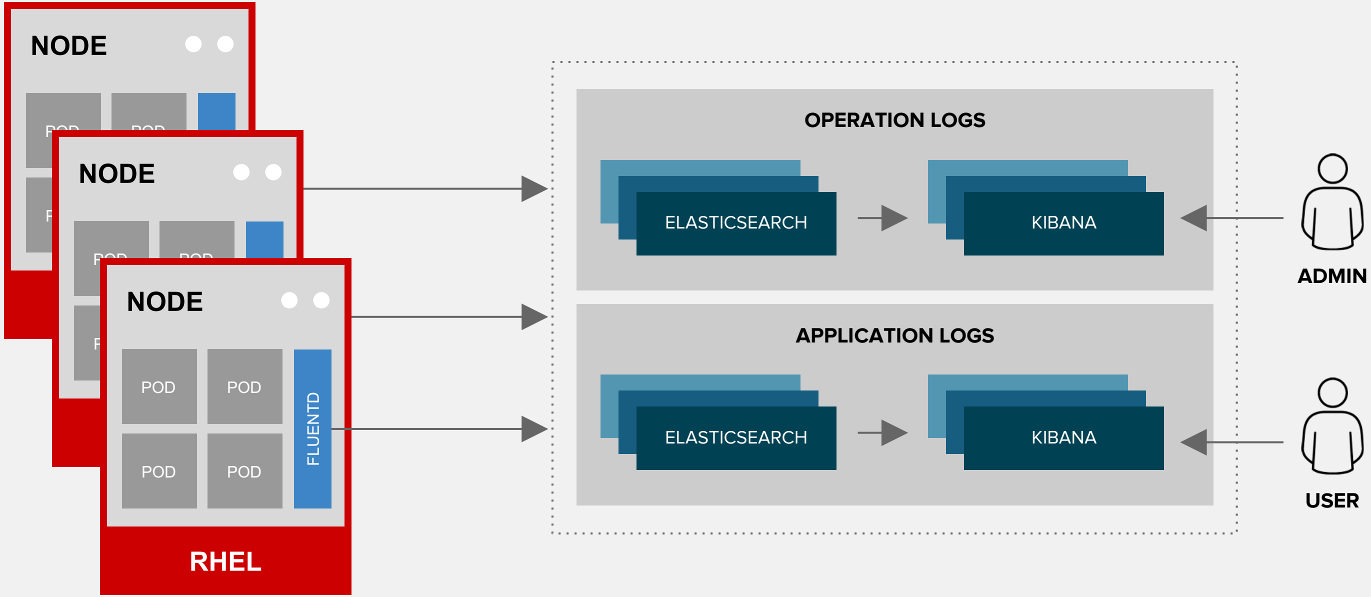
LOGGING & METRICS

CENTRAL LOG MANAGEMENT WITH EFK

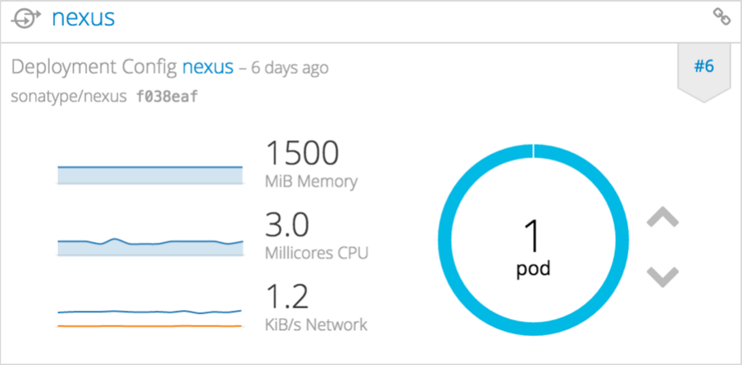
- EFK stack to aggregate logs for hosts and applications
  - **Elasticsearch:** an object store to store all logs
  - **Fluentd:** gathers logs and sends to Elasticsearch.
  - **Kibana:** A web UI for Elasticsearch.
- Access control
  - Cluster administrators can view all logs
  - Users can only view logs for their projects
- Ability to send logs elsewhere
  - External elasticsearch, Splunk, etc



CENTRAL LOG MANAGEMENT WITH EFK



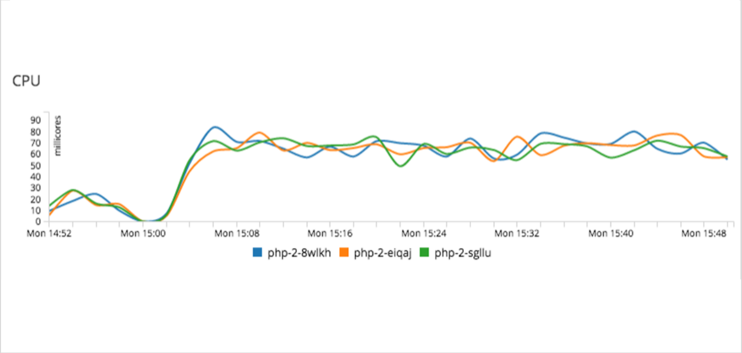
CONTAINER METRICS

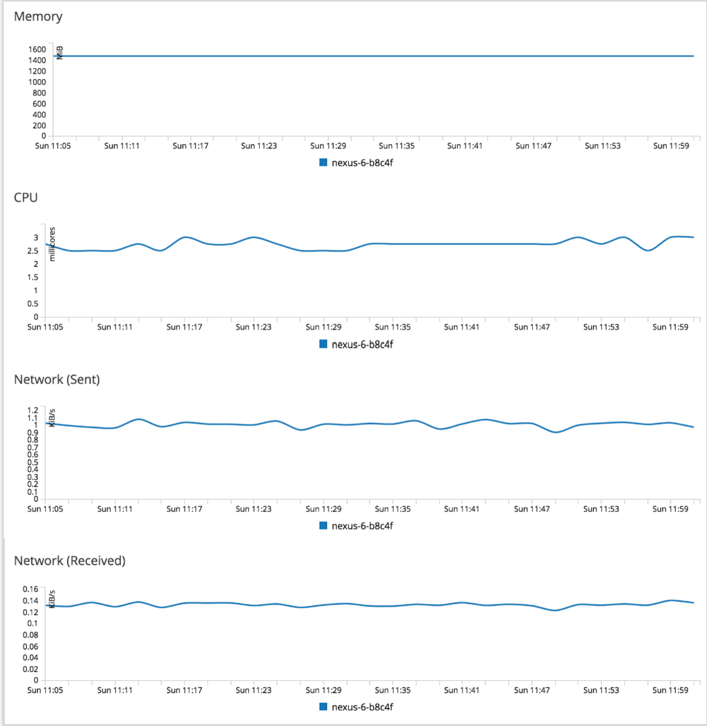


nexus

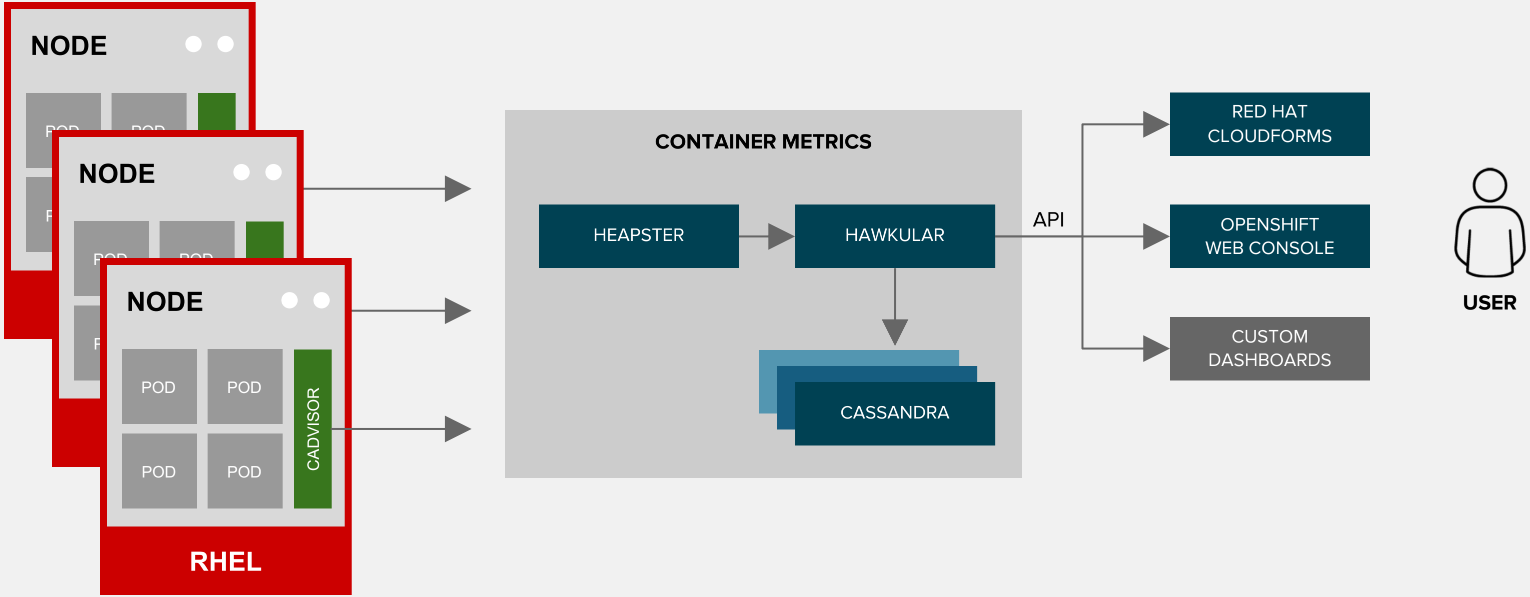
Deployment Config nexus – 6 days ago

sonatype/nexus f038eaf





CONTAINER METRICS



PERSISTENT STORAGE

PERSISTENT STORAGE

- Persistent Volume (PV) is tied to a piece of network storage
- Provisioned by an administrator (static or dynamically)
- Allows admins to describe storage and users to request storage

NFS

GlusterFS

OpenStack Cinder

Ceph RBD

AWS EBS

GCE Persistent Disk

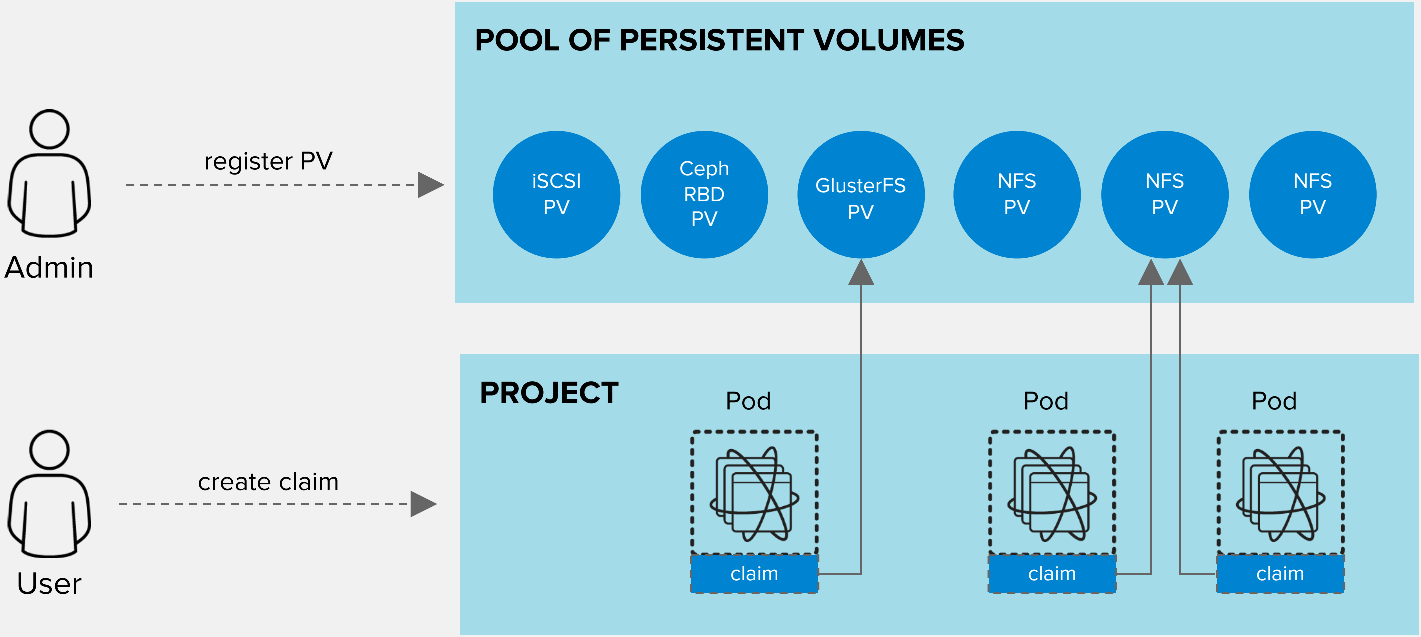
iSCSI

Fibre Channel

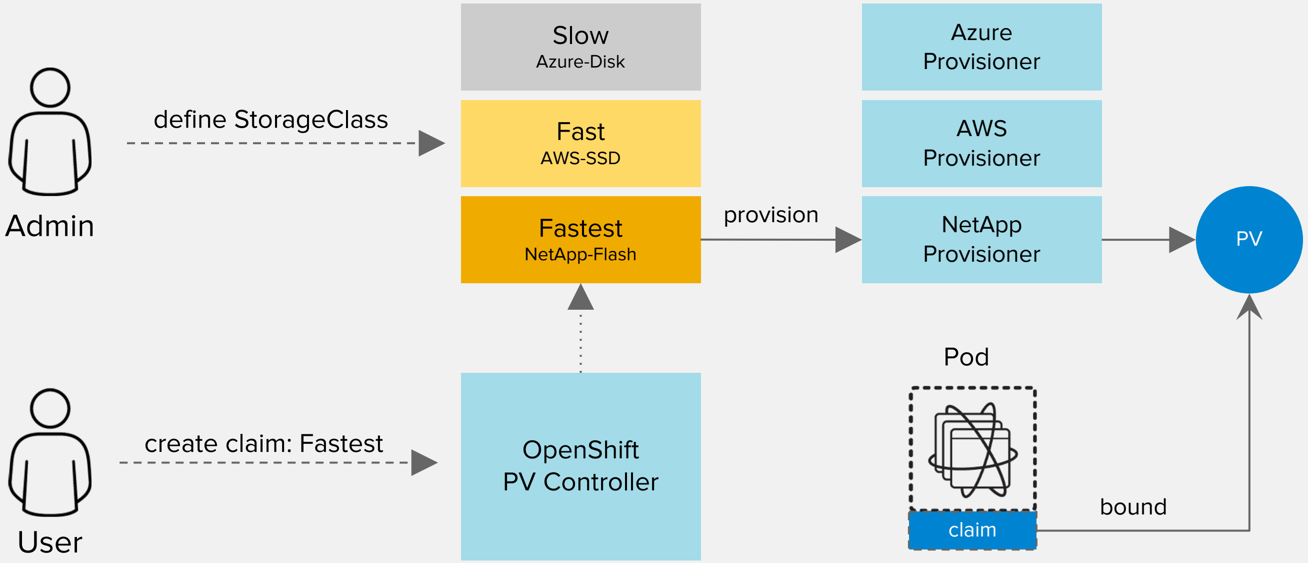
Azure Disk

Azure File

PERSISTENT STORAGE

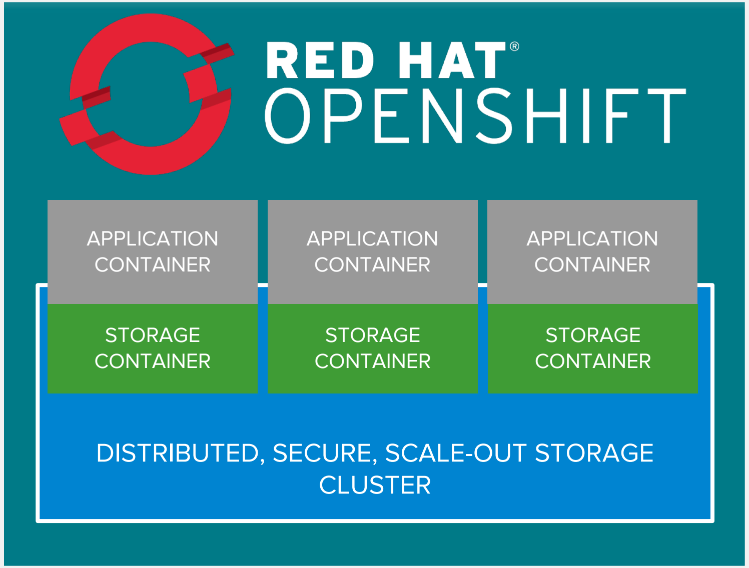


DYNAMIC VOLUME PROVISIONING



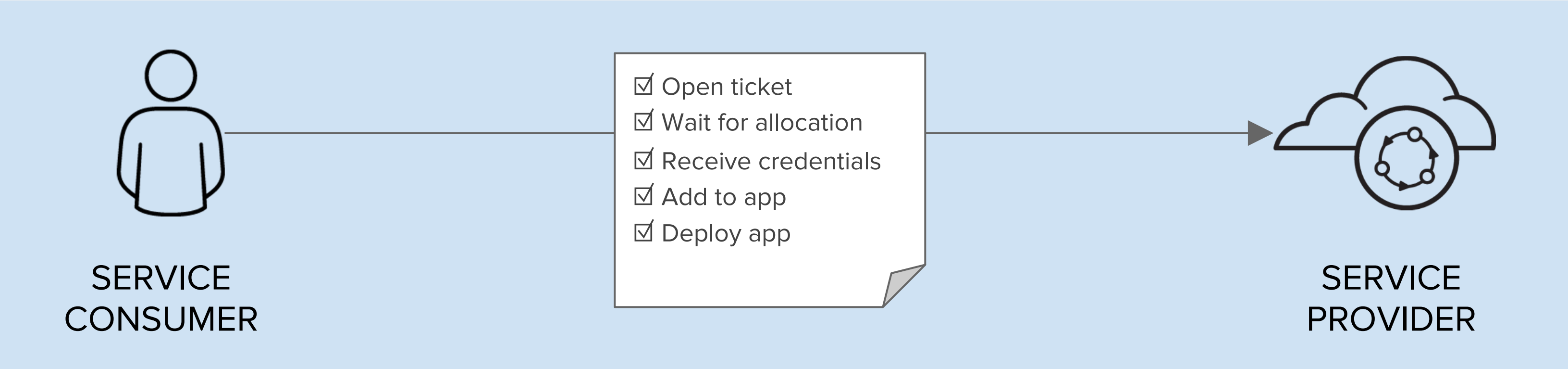
CONTAINER-NATIVE STORAGE

- Containerized Red Hat Gluster Storage
- Native integration with OpenShift
- Unified Orchestration using Kubernetes for applications and storage
- Greater control & ease of use for developers
- Lower TCO through convergence
- Single vendor Support



SERVICE BROKER

WHY A SERVICE BROKER?



SERVICE CONSUMER

☑ Open ticket
☑ Wait for allocation
☑ Receive credentials
☑ Add to app
☑ Deploy app

SERVICE PROVIDER

Manual, Time-consuming and Inconsistent

OPEN SERVICE BROKER API™

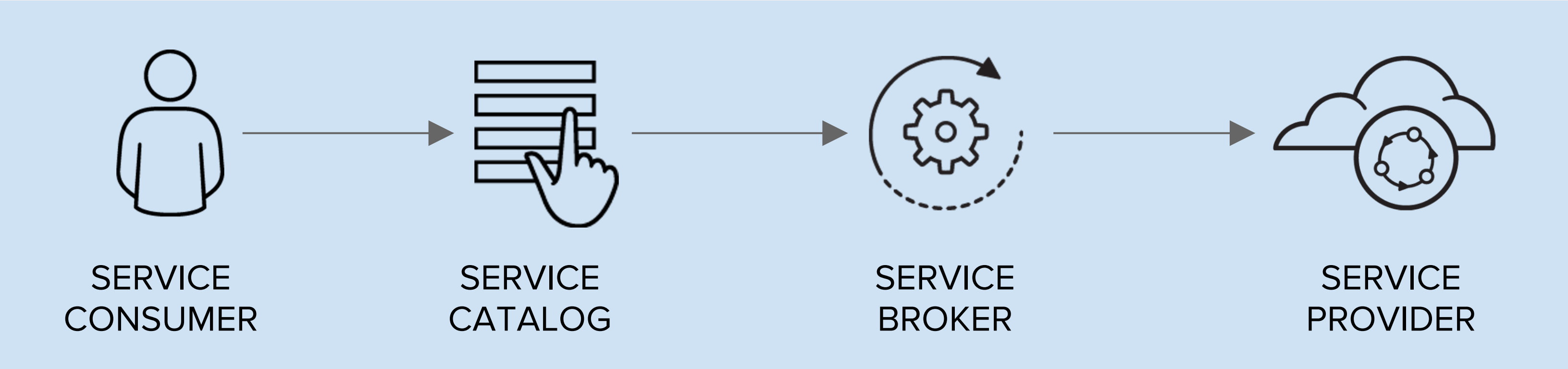
A multi-vendor project to standardize how services are consumed on cloud-native platforms across service providers

FUJITSU Pivotal

IBM redhat

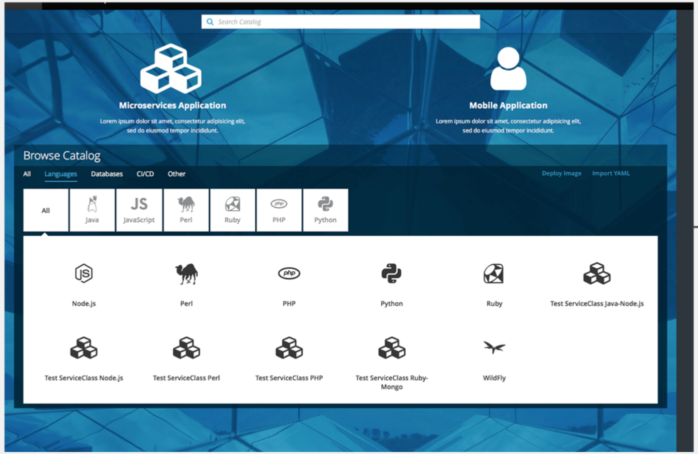
Google SAP®

WHAT IS A SERVICE BROKER?



Automated, Standard and Consistent

OPENSHIFT SERVICE CATALOG



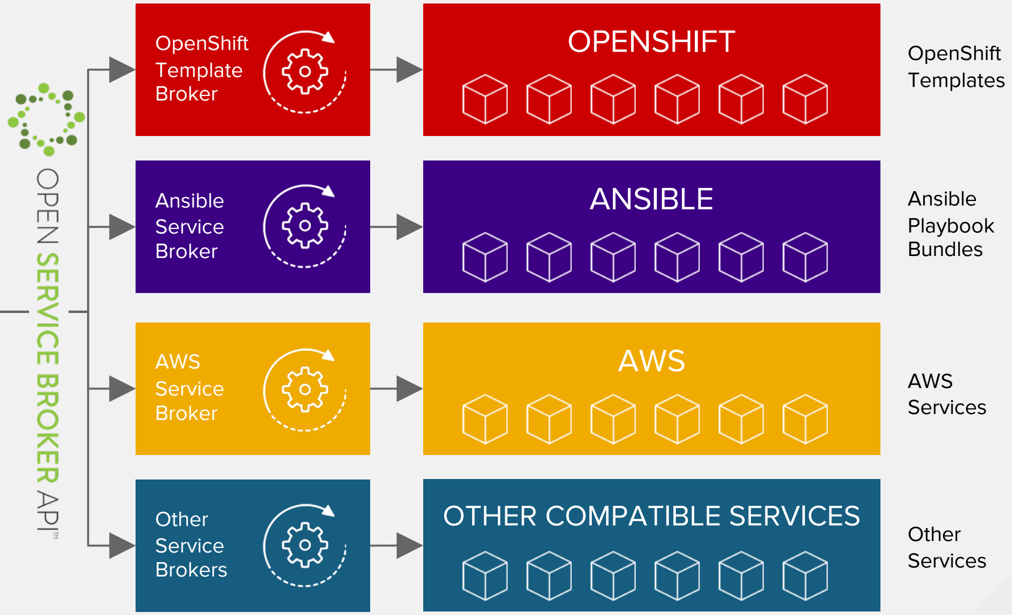
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt.

OPENSHIFT SERVICE CATALOG

OCP 3.6 TECH PREVIEW

OCP 3.7 GA



AWS SERVICE BROKER

- Targets Top 10 AWS Services
- Uses Ansible Playbook Bundles
- Available in OpenShift 3.7

SQS	SNS
RDS	EMR
DynamoDB	SNS
Lambda	RedShift SES
S3	ELB

OpenShift Demo

Patrick Cunning

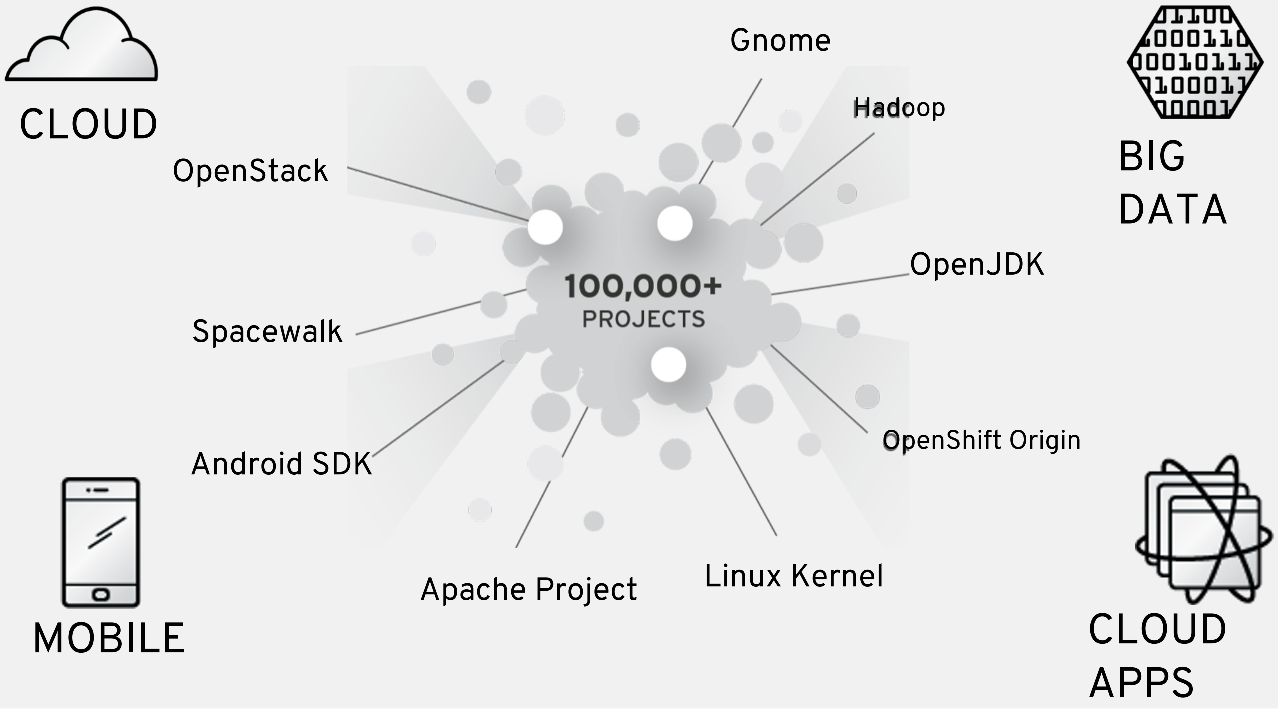
RHEL and Container Security Overview

Calvin Smith

Security and Compliance

Red Hat’s Objective:
To deliver the requisite security foundation our customers' infrastructure, applications, and workloads by providing technology that can be trusted, secured, and compliant.

New trends in IT...



CLOUD

OpenStack

Spacewalk

Android SDK

MOBILE

Gnome

Hadoop

OpenJDK

OpenShift Origin

Apache Project

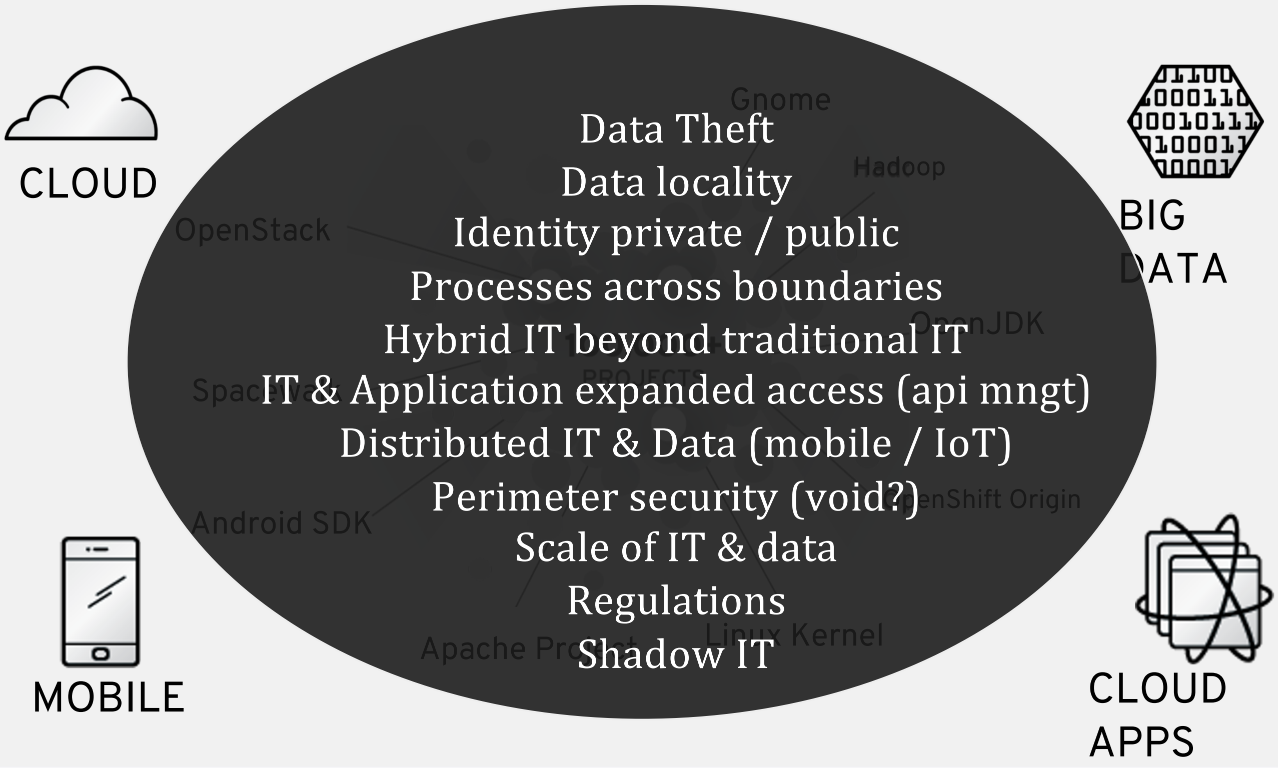
Linux Kernel

100,000+
PROJECTS

BIG
DATA

CLOUD
APPS

... bring New IT Security Challenges



Data Theft
Data locality
Identity private / public
Processes across boundaries
Hybrid IT beyond traditional IT
IT & Application expanded access (api mngt)
Distributed IT & Data (mobile / IoT)
Perimeter security (void?)
Scale of IT & data
Regulations
Shadow IT

“Most breaches we become aware of are caused by failure to update software components that are known to be vulnerable for months or even years,”

- René Gielen, vice president of Apache Struts

IT Governance

Provide Users/Business Users a governed IT

- Consumption control : CMP (Cloud Management Platform) : CloudForms
  - Which resources to access (private / public – high SLA / low SLA)
  - Service catalog in self-service mode
  - Hybrid Cloud governance : consumption control

- Business users : In-apps process and rules engine : JBoss BRMS / BPMS

- Devs : Tooling for better collaboration (devops) : Openshift

- Ops : get to scale
  - Standardized platform : controlled deployment profiles
  - Multi-tenants platforms Openshift, Openstack, Satellite

Software Quality

Mitigating risks on software is inherent in Open source community Development

- At Red Hat , we add :
  - Q&A
  - Build processes
  - Cryptographic signatures
  - Signed Protected access (ssl)
  - Security teams
  - Maintain for 10 years a stable and secure baseline (backporting)
- Makes RH a trusted supplier to customers

Certifications

Business regulations

Government

- FIPS-140-2 (cryptographic implement properly)
- USGV6 (DoD IPV6 requirements)
- DISA STIG (Secure Technical implementation Guidelines)
- FISMA (Federal Information Security Management Act)
- FedRAMP (variant of FISMA process for cloud providers)
- US Army certificate of Networthiness
- USGCB : US government Configuration Baseline

Finance

- PCI DSS 3.0
- SOX 404

COMMON CRITERIA CERTIFICATION
Security Enhanced Linux (SELinux)
Mandatory Access Control between virtual machines

Technical regulations

SCAP : Secure Content Automation Protocol (configuration requirements)

OVAL : Open Vulnerability and Assessment Language : to describe vulnerabilities (founded by RH in 2002)

CVE : common Vulnerability Enumeration : common identifier for common flaws

IAVAs : (Information Assurance Vulnerability Alerts) similar to CVE for DOD personnel

Common Criteria : up to EAL 4+

PCI DSS v3 : eg : Establish a process to identify security vulnerabilities, using reputable outside sources for security vulnerability information, and assign a risk ranking (for example, as “high,” “medium,” or “low”) to newly discovered security vulnerabilities.*

SOX 404 :eg : Patch and configuration management to ensure that financial data is protected and that there is an audit trail to documenting all changes

Red Hat Security team

- Monitoring vulnerabilities, exploits & threats
- Triage
- Escalation and troubleshooting through lifecycle
- Communication with other affected vendors
- Internal communication, documentation, advisory
- Responsible for errata release
- Metrics and feedback to engineering
- Single point of contact for customers

EXCEPTIONAL SECURITY

95

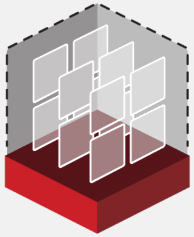
% of all critical security issues in Red Hat Enterprise Linux are fixed within…

1

day.

CONTAINERS

LINUX IS CONTAINERS
CONTAINERS ARE LINUX



CONTAINERS

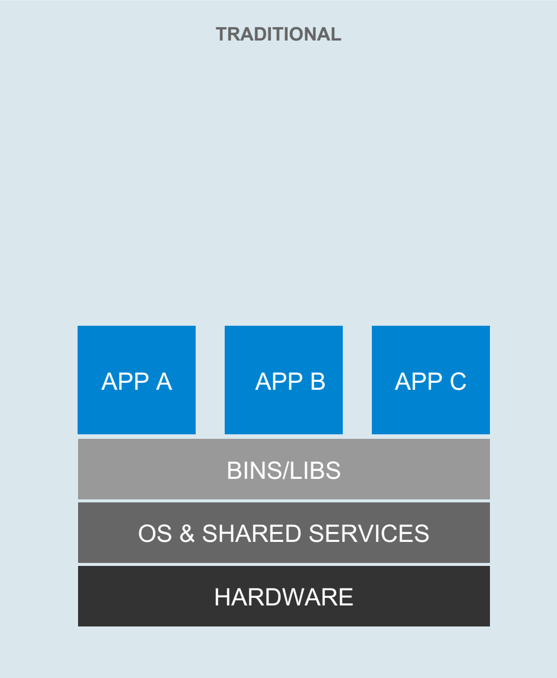
- Next-generation application platform for existing and new apps
- Portable across the hybrid cloud
- Container companies must be Linux companies

RHEL HOST
Server Operating Environment

KERNEL	DEVICE SUPPORT	SYSTEMD	SELINUX	NAMESPACE	CGROUPS

PHYSICAL	VIRTUAL	PRIVATE	PUBLIC

Evolution: Traditional Enterprise OS



Traditional application deployment

- Single userspace runtime shared between applications.
- Environment and life cycle defined by host OS.
- Trend to isolate apps on hardware level.
- Managed by IT, very limited delegation.
- Stable, long maintenance, few updates, hardware-centric.
- Very limited flexibility.
- Resources generally underutilized.

New project

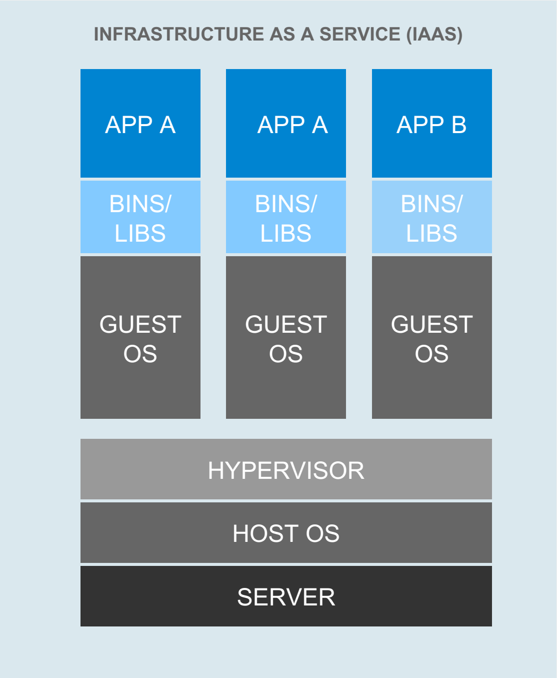
Application dependency

Application rollout

Security Fix

OS Version Update

Evolution: Virtualization & IaaS



Application deployment via virt & IaaS

- Application isolation per VM.
- Guest environment and lifecycle defined by application.
- Application and runtime abstracted from hardware.
- Higher flexibility at cost of increased redundancy and overhead.
- Complex multi-level management of host and VM layers
- Delegation along the Host / VM boundary.

New project

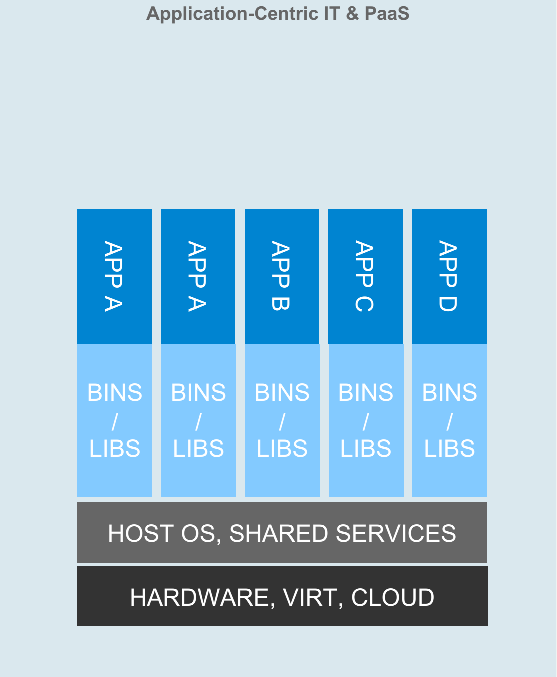
Application dependency

Application rollout

Security Fix

OS Version Update

Evolution: Application-Centric IT



App delivery using Docker containers

- Application packaged with individual runtime stack using Docker and deployed into containers.
- Multi-instance, multi-version, maximal flexibility, minimal overhead.
- Delegation along the container boundaries.
- Shared services provided by host / container environment.
- Standardized hardened container host, clustering, orchestration.

New project

Application dependency

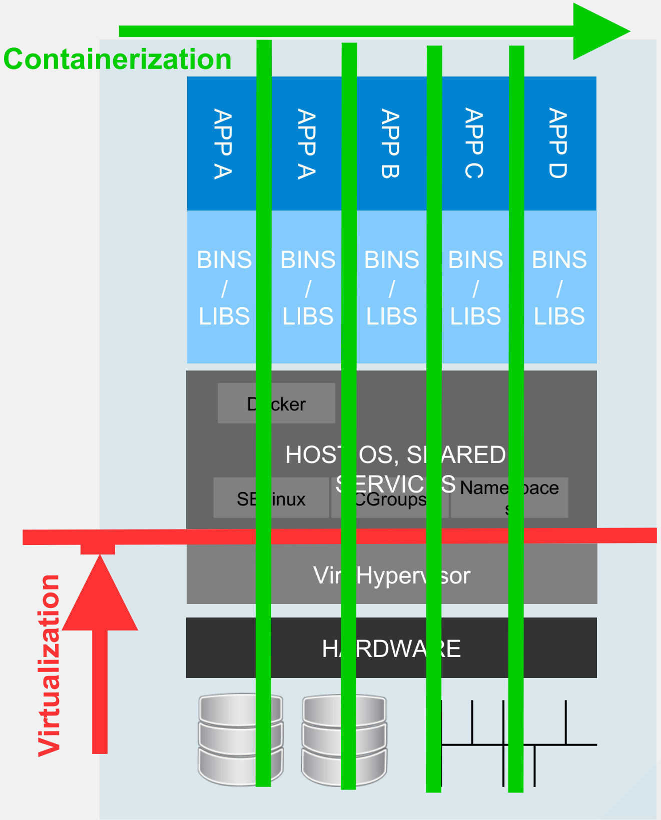
Application rollout

Security Fix

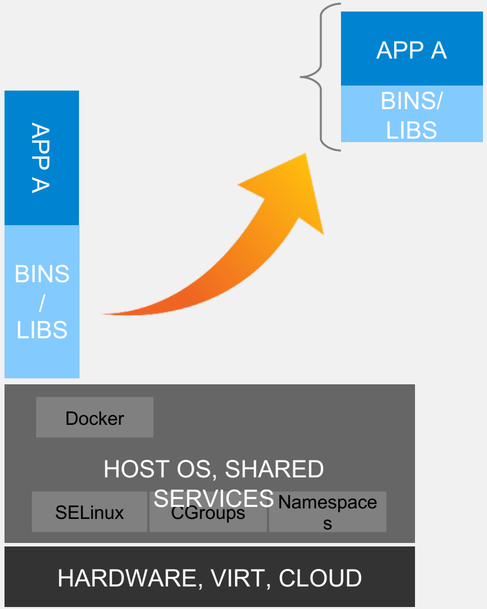
OS Version Update

Containers vs. Virt?

- Generally complementary concepts
  - Virtualization: vertical abstraction
  - Containerization: horizontal segmentation
- Containers used to replace virtualization where container paradigms more applicable:
  - Horizontal application isolation
  - Lightweight delegation
  - “Application Virtualization”
  - Density
- Containers on top of Virt/Cloud common.



Tech Details – Containers & Docker



- Linux Containers are a combination of kernel features:
  - namespaces
  - SELinux
  - control groups
- Containers provide lightweight isolation of process, network, filesystem spaces.
- Docker builds on Linux Containers, adds an API, an image format and a delivery and sharing model.

Key elements of Linux Containers

- Process Isolation
  - kernel namespace
- Security
  - SElinux
- Resource Management
  - cgroups
- Container Management
  - Docker

Process Isolation - Namespaces

- Namespaces isolates and limits the visibility that processes have of system resources
- Create a new environment with a subset of the resources
- Once set up, namespaces are transparent for processes
- Can be used in custom and complex scenarios

Process Isolation - Namespaces

Namespaces	Functionality	What does it mean?
Mount	Isolate the set of FS mount points seen by processes	/tmp in container can be different in ns' Remount '/' read only within namespace
PID (process ID)	Process can have same PID in different NS (include PID1)	Process in NS can't see/interact with process outside All processes are visable in 'root' PID NS
Network	Isolate the networking stack: ip addr, routes, netfilter iptable rules	Each NS has its own private loopback IF Commonly used with virtual ethernet IF pair
UTS	Set a different host and domain names for NS	No impact to the rest of the system Useful when combined with Network NS
IPC	Private inter-process communication environment: message queues, semaphores, shared memory	Resources are only accessible within the Namespace

SELinux

- Where did it come from?
  - Created by the United States National Security Agency (NSA) as set of patches to the Linux kernel using Linux Security Modules (LSM)
  - Released by the NSA under the GNU General Public License (GPL) in 2000
  - Adopted by the upstream Linux kernel in 2003

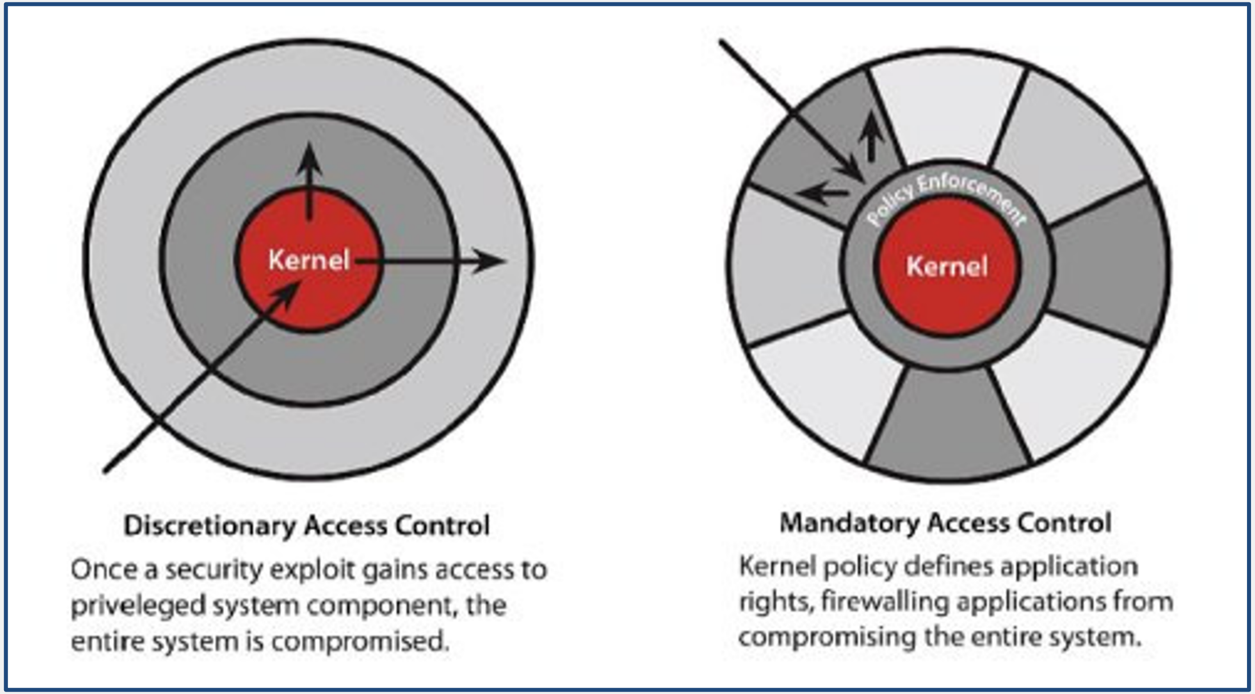
Discretionary Access Control (DAC)

- Historically, Linux and Unix systems have used discretionary access control.
  - Ownership (user, group, and other) plus permissions.
  - Users have the ability (discretion) to change permissions on their own files. A user can chmod +rwx his or her home directory, and nothing will stop them. Nothing will prevent other users or processes from accessing the contents of his home directory.
- Traditionally, the root user is omnipotent

Mandatory Access Control (MAC)

- On a mandatory access control system, there is policy which is administratively set and fixed.
- Even if you change the DAC settings on your home directory, if there is a policy in place which prevents another user or process from accessing it, you're generally safe.

DAC vs. MAC



Discretionary Access Control

Once a security exploit gains access to priveleged system component, the entire system is compromised.

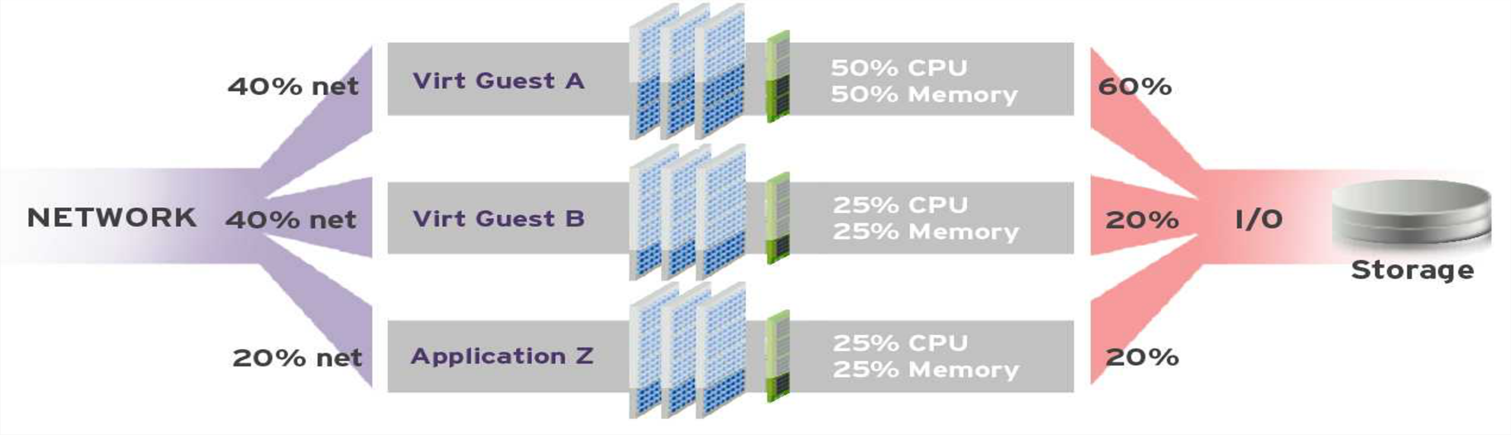
Mandatory Access Control

Kernel policy defines application rights, firewalling applications from compromising the entire system.

So How Does SELinux Work?

- Everything is labelled
  - Files, processes, ports, etc., are all labeled with an SELinux context.
  - For files and directories, these labels are stored as extended attributes on the filesystem.
  - For processes, ports, etc., the kernel manages these labels.
- SELinux policies define what is allowed
- What is not allowed is denied

Strong Resource Control with CGroups

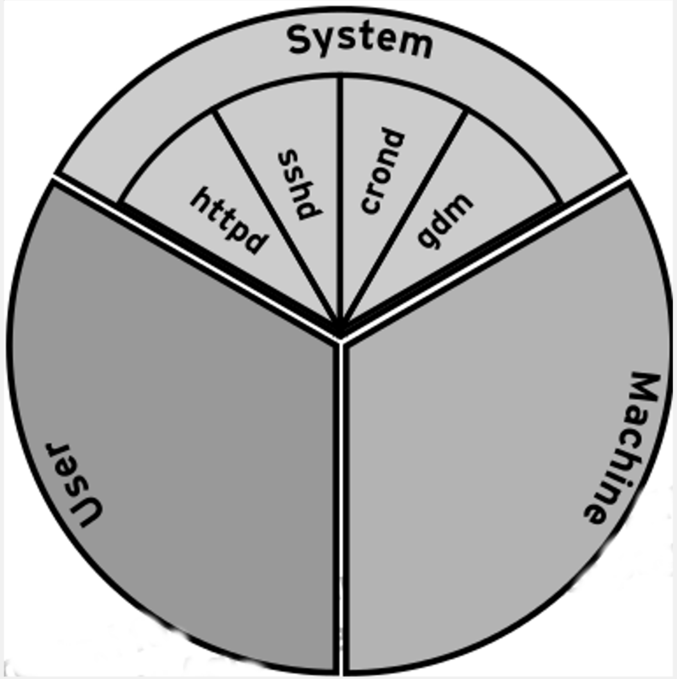


Resources managed by cgroups

cpuset cpuacct cpu freezer	CPU	Memory	memory hugetlb
blkio devices	Storage	Network	net_cls

Control Groups

When using cgroups, resources are placed in controllers representing the type of resource; for example, cpu for CPU time, memory for memory usage, and blkio for disk I/O. Cgroups can contain multiple controllers. A cgroup is then split into slices, like a pie chart.



Introduction to SCAP

Shawn Wells

Everyone knows that SCAP is a suite of XML standards for creating automated checklists for configuration and vulnerability scans!

SCAP → HTML

SCAP → HTML

OpenSCAP → Firefox

GUIDANCE

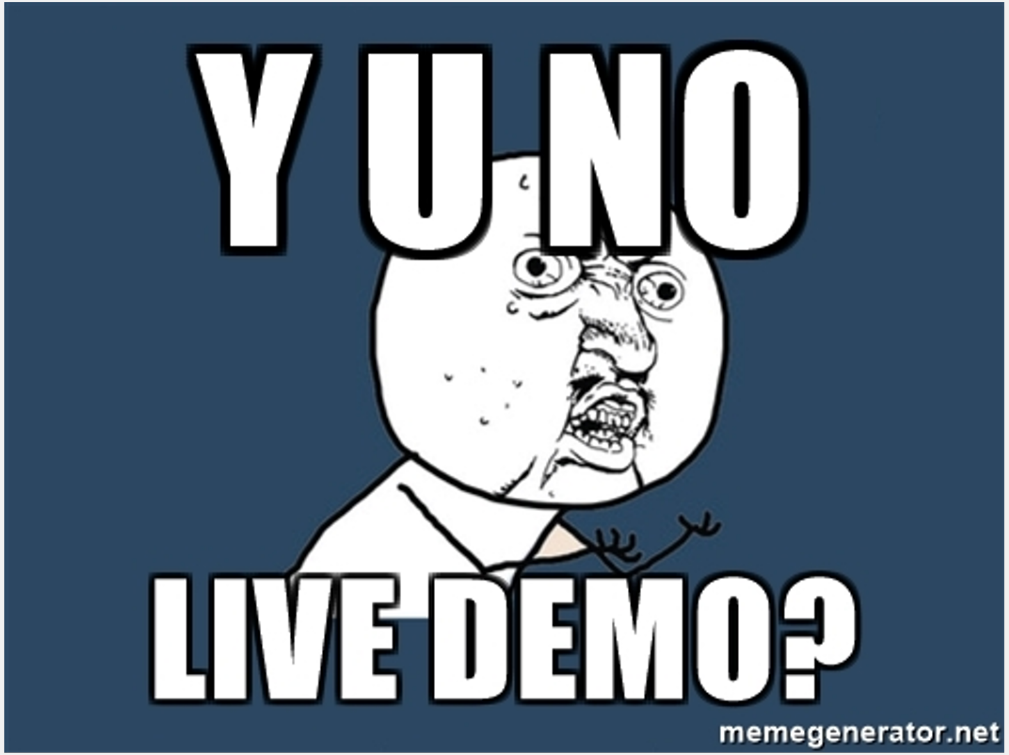
GUIDANCE	
VERIFICATION	

GUIDANCE	
VERIFICATION	
REMEDIATION	

GUIDANCE	XCCDF
VERIFICATION	
REMEDIATION	

GUIDANCE	XCCDF
VERIFICATION	OVAL
REMEDIATION	

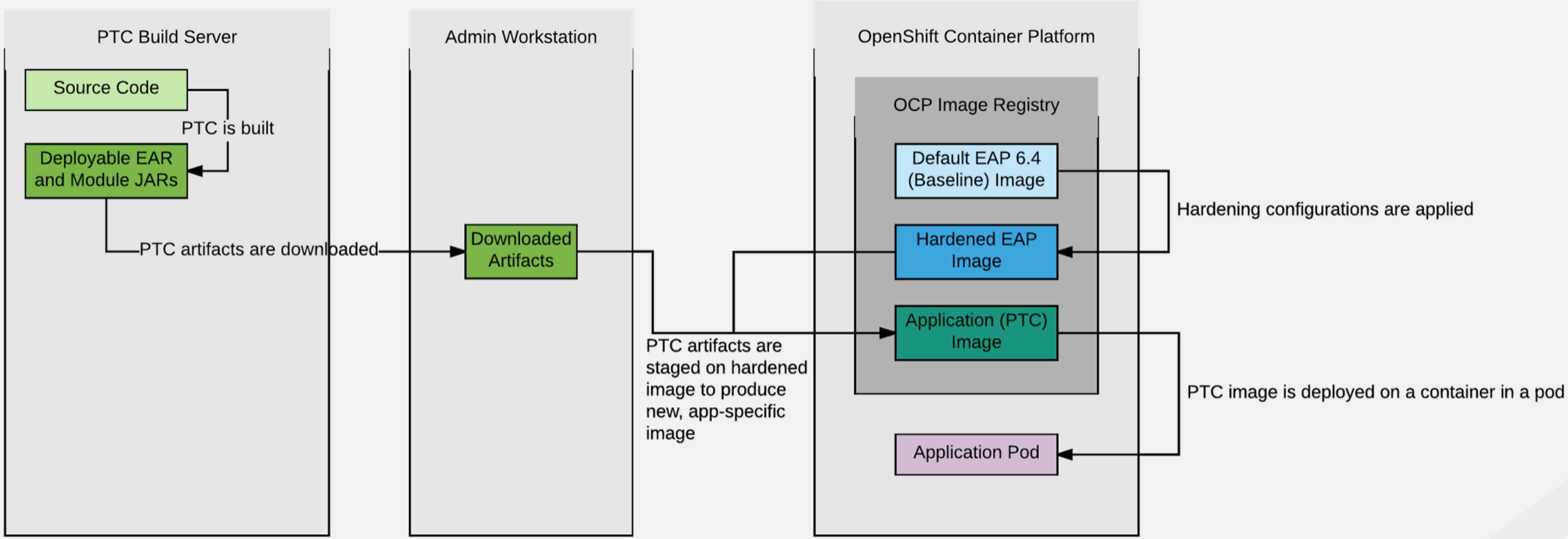
GUIDANCE	XCCDF
VERIFICATION	OVAL
REMEDIATION	<fix> scripts (ansible, bash…)


Y U NO
LIVE DEMO?
memegenerator.net

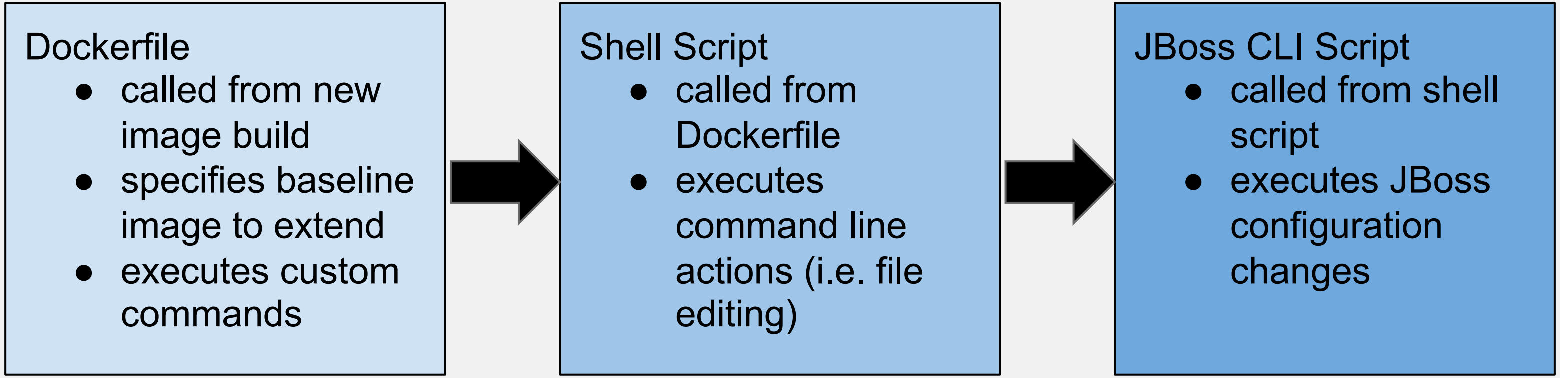
Java Source Code to Hardened Container Image

Khary Mendez

Current Pipeline -
Source Code to Live Container


PTC Build Server
Source Code
PTC is built
Deployable EAR and Module JARs
PTC artifacts are downloaded
Admin Workstation
Downloaded Artifacts
PTC artifacts are staged on hardened image to produce new, app-specific image
OpenShift Container Platform
OCP Image Registry
Default EAP 6.4 (Baseline) Image
Hardened EAP Image
Application (PTC) Image
Application Pod
Hardening configurations are applied
PTC image is deployed on a container in a pod

EAP Hardening



Dockerfile

- called from new image build
- specifies baseline image to extend
- executes custom commands

Shell Script

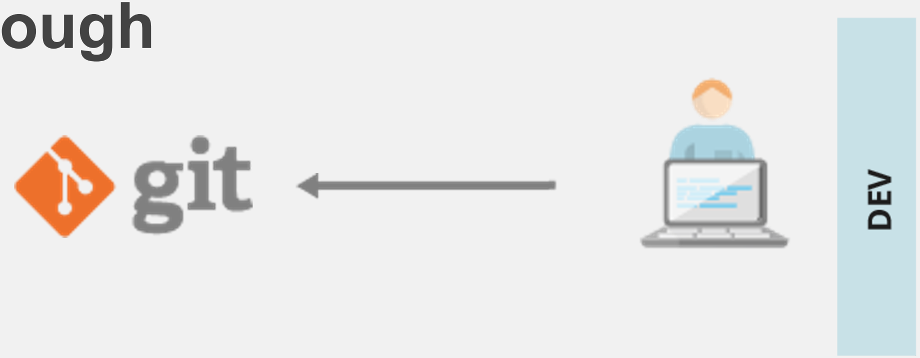
- called from Dockerfile
- executes command line actions (i.e. file editing)

JBoss CLI Script

- called from shell script
- executes JBoss configuration changes

Demo

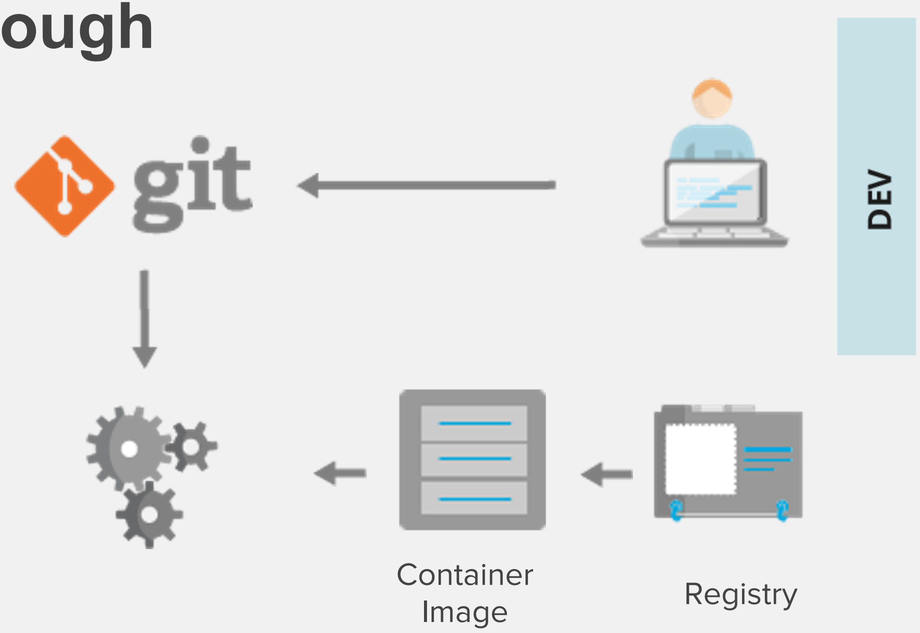
Source 2 Image Walk Through



Code

Developers can leverage existing development tools and then access the OpenShift Web, CLI or IDE interfaces to create new application services and push source code via GIT. OpenShift can also accept binary deployments or be fully integrated with a customer’s existing CI/CD environment.

Source 2 Image Walk Through



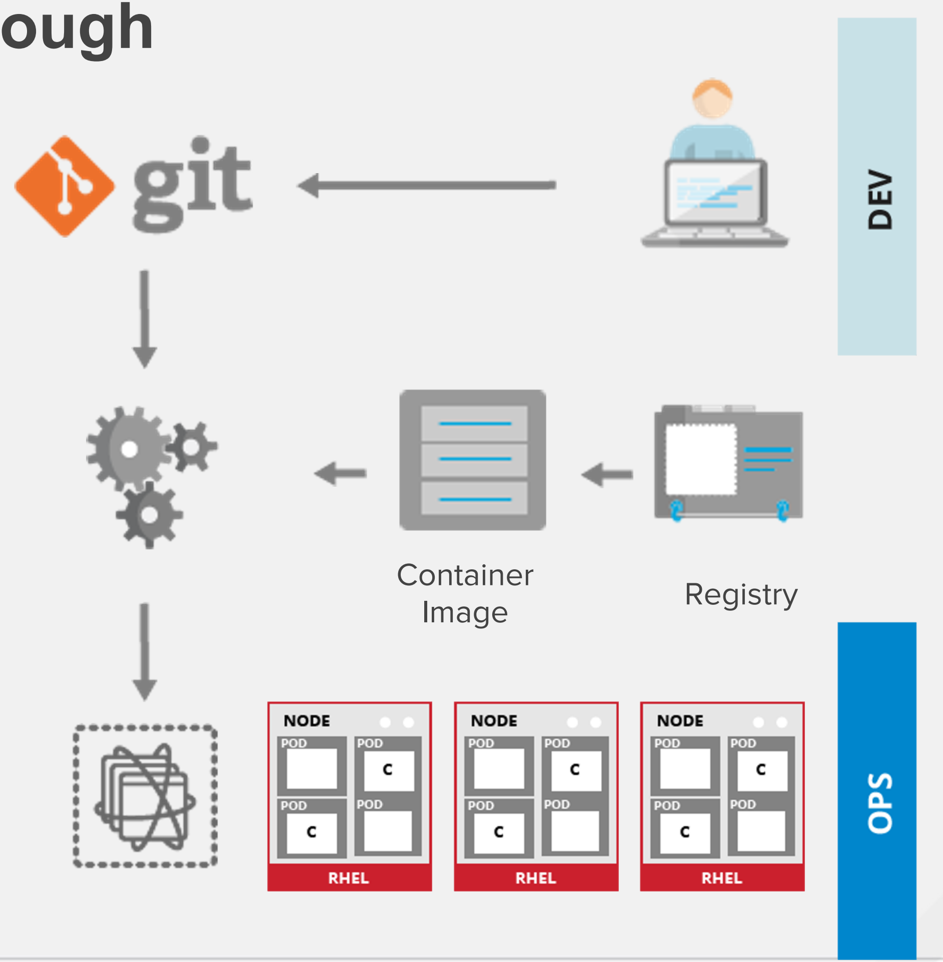
Build

OpenShift automates the Docker image build process with Source-to-Image (S2I). S2I combines source code with a corresponding Builder image from the integrated Docker registry. Builds can also be triggered manually or automatically by setting a Git webhook. Add in Build pipelines

Source 2 Image Walk Through

Deploy

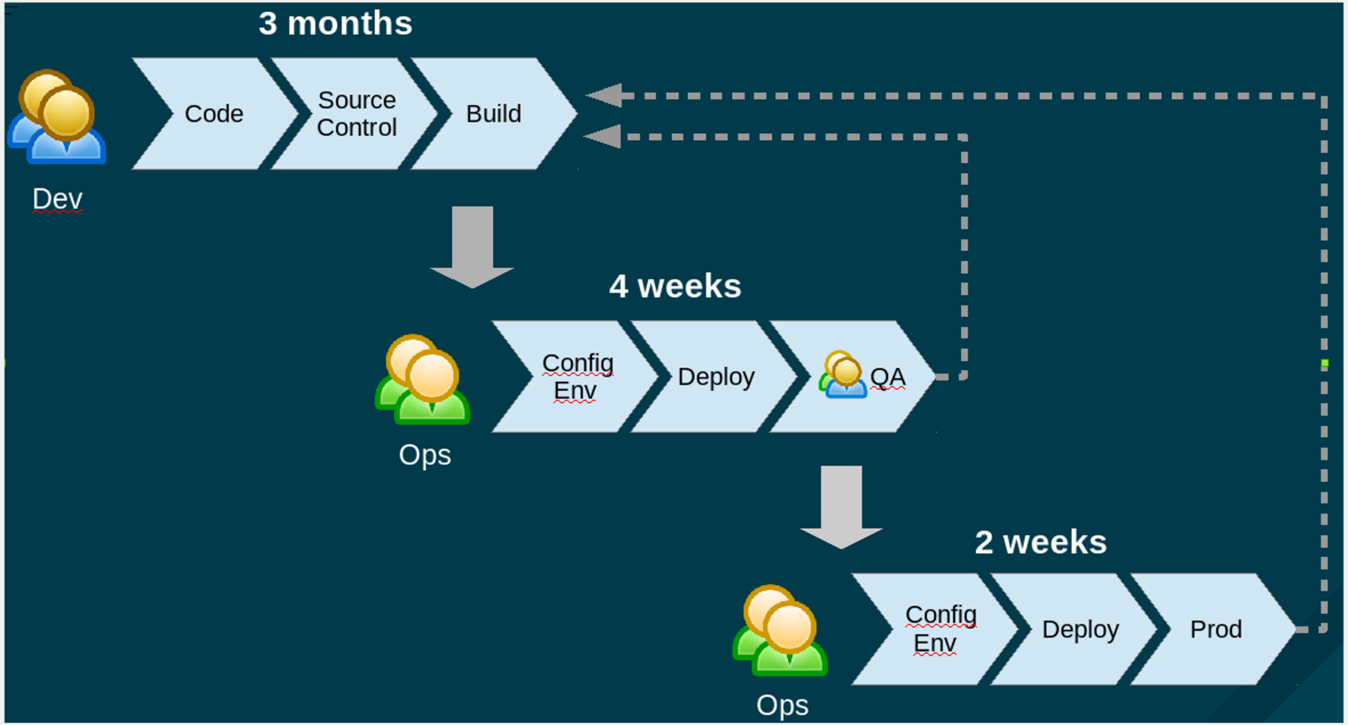
OpenShift automates the deployment of application containers across multiple Node hosts via the Kubernetes scheduler. Users can automatically trigger deployments on application changes and do rollbacks, configure A/B deployments & other custom deployment types.



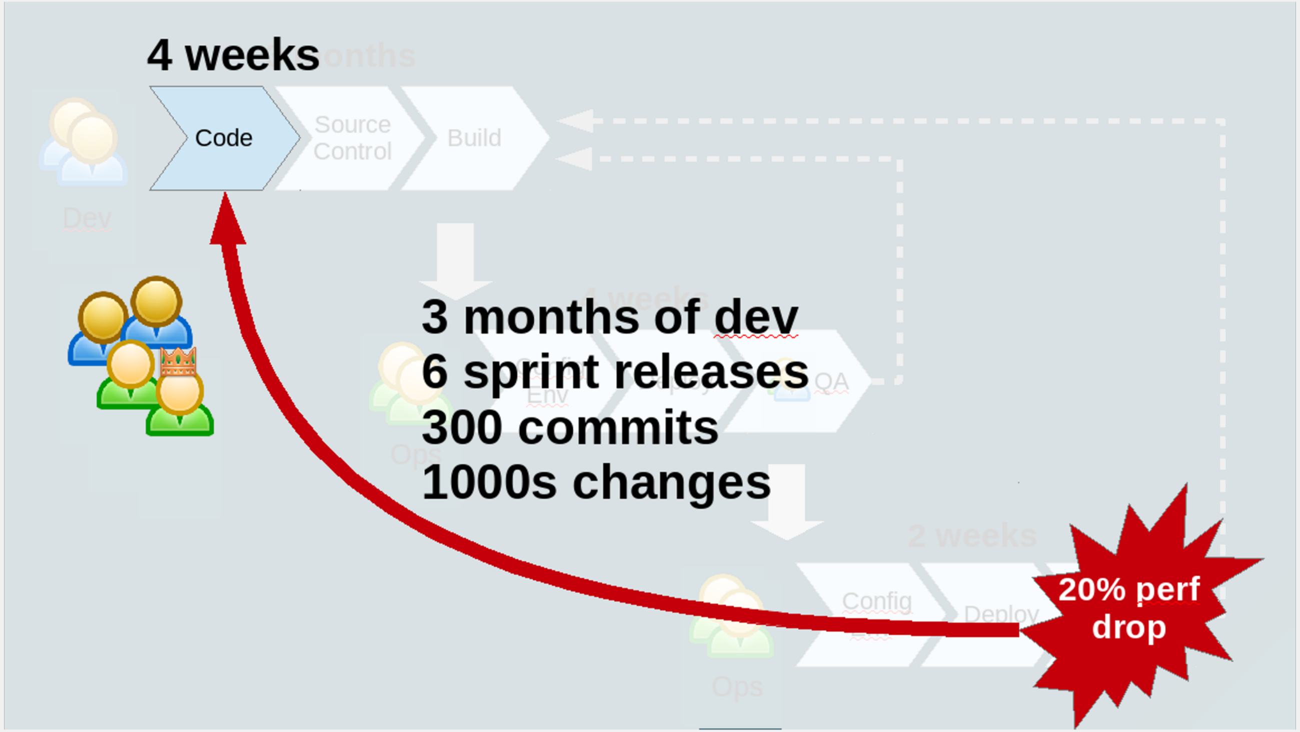
Continuous Integration and Continuous Delivery Pipeline

Khary Mendez

Day in The Life of a Developer (Old Way)



Day in The Life of a Developer (Old Way)

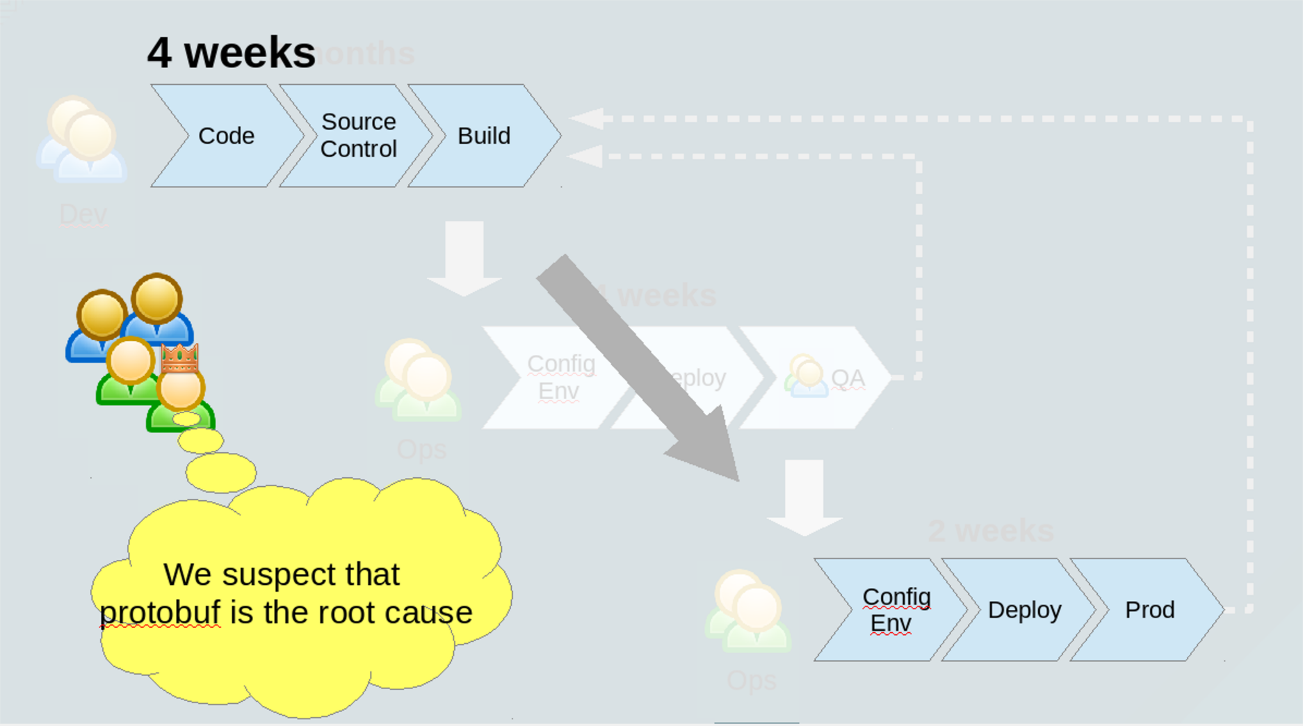


4 weeks

3 months of dev
6 sprint releases
300 commits
1000s changes

20% perf drop

Day in The Life of a Developer (Old Way)

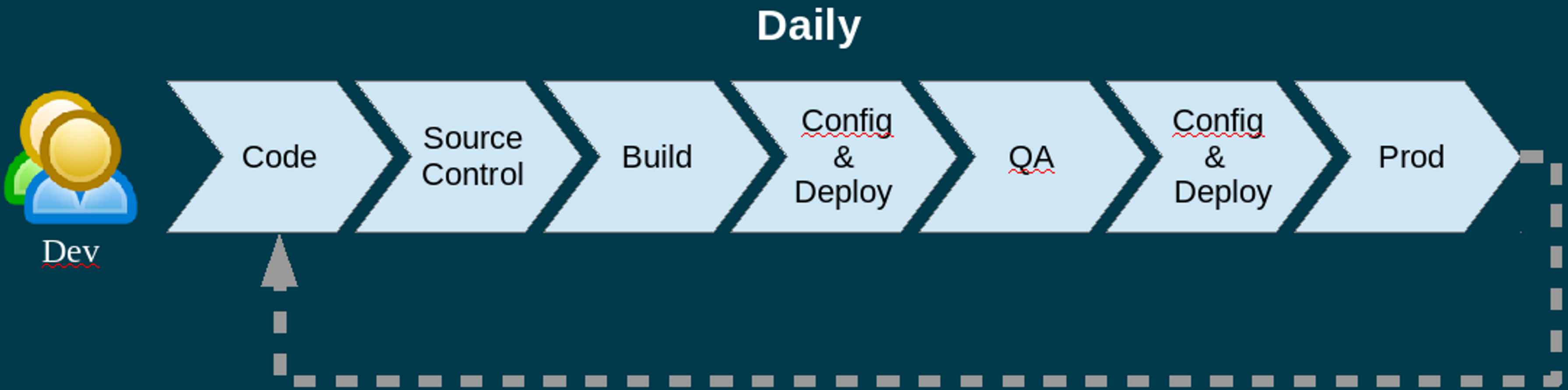


Business Problems

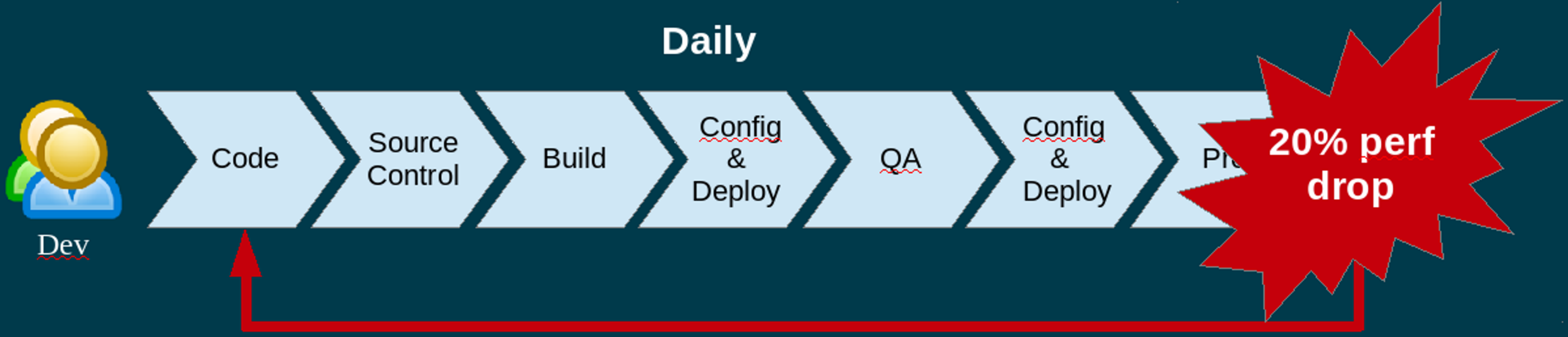
Due to constraints, environments aren't identical and the software is brittle.

- Cost - Issues are expensive to fix
- Security- Often left to the end
- Administrative - Overhead in provisioning environments
- People / Skillset - Reliance on key people

The New Way

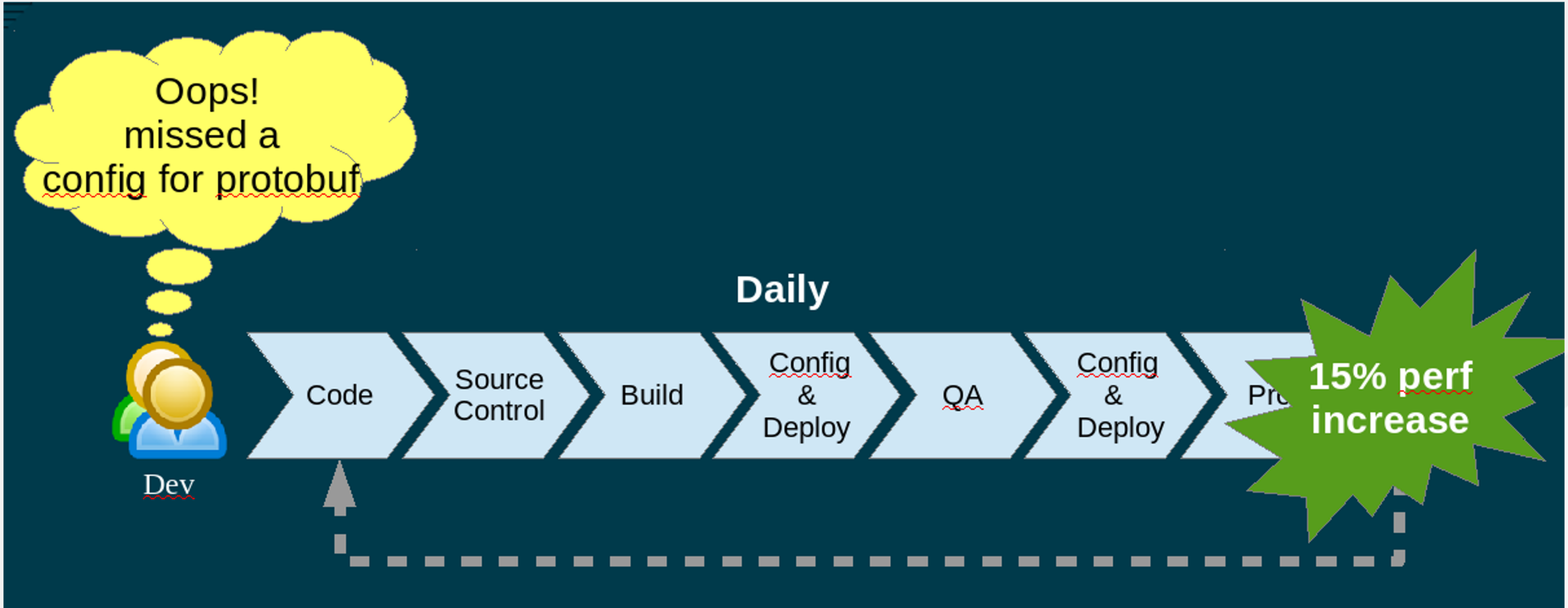

Daily
Dev
Code
Source Control
Build
Config & Deploy
QA
Config & Deploy
Prod

The New Way

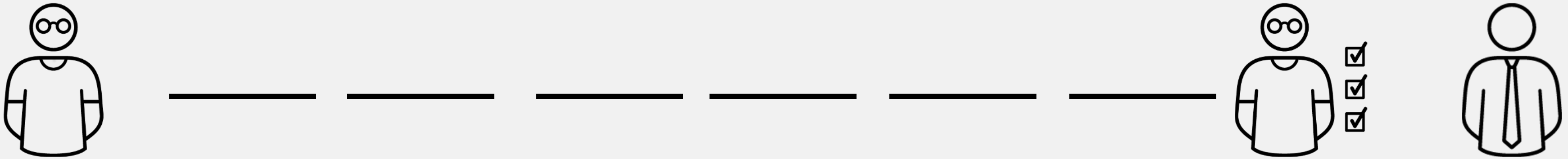


10 commits

The New Way


Oops!
missed a
config for protobuf
Dev
Daily
Code
Source Control
Build
Config & Deploy
QA
Config & Deploy
Pr
15% perf increase

Demo CI/CD Pipeline



Developers

Test

Management

Checkout	Build	Scan Code	Archive	Create Image	Scan Image	Deploy to Dev	Deploy to Test	Deploy to Prod

Gogs A painless self-hosted Git service.

Maven™

sonarqube

Sonatype

moby project

OpenSCAP

OPENSHIFT® by Red Hat®

Demo Scenario

- Developer pushes code change into the Git repository (GOGS)
- A security vulnerability is found in the code
- Developer fixes code
- A CVE is detected in the EAP image
- Image is updated
- Pipeline succeeds, updated code is deployed, then promoted to test

Scheduling OSCAP Scans

Cameron wyatt

Open Control to Automate Security Authorization Package

Shawn Wells

CAN YOU EXPLAIN YOUR ENTIRE ATO PROCESS?

Thank You