



TEN LAYERS OF CONTAINER SECURITY

Tim Hunt
Kirsten Newcomer

May 2017

ABOUT YOU

Are you using containers?

What's your role?

- Security professionals
- Developers / Architects
- Infrastructure / Ops

Who considers security part of their job?

VALUE OF CONTAINERS

INFRASTRUCTURE

- Sandboxed application processes on a shared Linux OS kernel
- Simpler, lighter, and denser than virtual machines
- Portable across different environments

APPLICATIONS

- Package my application and all of its dependencies
- Deploy to any environment in seconds and enable CI/CD
- Easily access and share containerized components

WHY ARE WE HAVING
THIS CONVERSATION?



SECURING CONTAINERS: THE TOP TEN LIST

1. Container Host & Multi-tenancy
2. Container Content
3. Container Registries
4. Building Containers
5. Deploying Containers
6. Container Platform
7. Network Isolation
8. Storage
9. API Management
10. Federated Clusters

1

CONTAINER HOST & MULTI-TENANCY THE OS MATTERS

RED HAT ENTERPRISE LINUX



RED HAT ENTERPRISE LINUX ATOMIC HOST

THE FOUNDATION FOR SECURE, SCALABLE CONTAINERS

A stable, reliable host environment with built-in security features that allow you to isolate containers from other containers and from the kernel.

Minimized host environment tuned for running Linux containers while maintaining the built-in security features of Red Hat Enterprise Linux..

SELinux

Kernel namespaces

Cgroups

Capabilities

Seccomp

SECURITY FEATURES ON BY DEFAULT IN OPENSHIFT

2 CONTENT: USE TRUSTED SOURCES

- Are there known vulnerabilities in the application layer?
- Are the runtime and OS layers up to date?
- How frequently will the container be updated and how will I know when it's updated?

The screenshot shows the Docker Hub page for the 'Python 3.4 platform for building and running applications' image. The page title is 'Python 3.4 platform for building and running applications' by Red Hat, Inc. in Product Red Hat Enterprise Linux. The image is available at 'registry.access.redhat.com/rhsc1/python-34-rhe17' and was updated 7 days ago. The current version is 3.4-13.16, with a Health Index of 100%. The page has tabs for Overview, Get this image, Tech Details, Documentation, and Tags. The 'Tags' tab is active, showing a timeline of updates. The timeline shows three updates: 3.4-13.14 (RHBA-2017:0404), 3.4-13.15 (RHBA-2017:0975), and 3.4-13.16 (RHBA-2017:1127). The timeline is set against a background of a calendar grid for March and April 2017. A vulnerability grid on the right side of the page shows the status of various vulnerabilities (A-F) across different versions.

Red Hat rebuilds container images when security fixes are released

2 CONTENT: USE TRUSTED SOURCES

Standardization
makes security &
ops work easier

The screenshot displays the Red Hat Container Catalog interface. At the top, there is a navigation bar with the Red Hat logo, 'CUSTOMER PORTAL', and menu items for 'Products & Services', 'Tools', 'Security', and 'Community'. Below this is a search bar labeled 'Red Hat Container Catalog' with a 'SEARCH' button. The main content area is titled 'Enterprise-ready Containers' and includes the subtitle 'Your trusted source for secure, certified, and up-to-date container images'. There are two tabs: 'Recently Updated' (selected) and 'Recently Added'. Three container images are listed:

- RHMAP 4.3 Millicore**: Provides the RHMAP Millicore Server. Note: This is a protected repo and you need entitlements to RHMAP SKU to access the images. Last update: 2 days ago. Health index: A. Pulls: 184.
- RHMAP 4.3 MySQL 5.5**: Provides an extension to the RHSC MySQL Docker image for RHMAP. Note: This is a protected repo and you need entitlements to RHMAP SKU to access the images. Last update: 2 days ago. Health index: A. Pulls: 171.
- RHMAP 4.3 MongoDB 3.2**: Provides an OpenShift Container Platform MongoDB 3.2 container for RHMAP, based on rhsc/mongodb-32-rhel7. Note: This is a protected repo and you need entitlements to RHMAP SKU to access the images. Last update: 2 days ago. Health index: A. Pulls: 279.

Developers want
latest & greatest for
best features

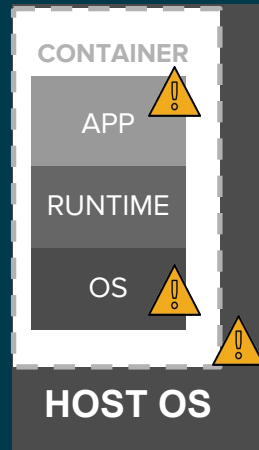
Consider breadth and diversity of your software content

3

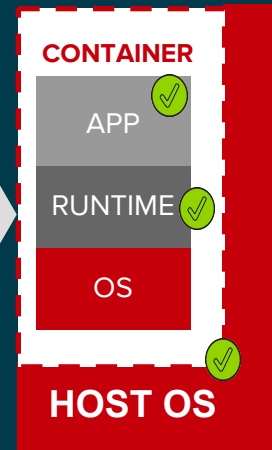
PRIVATE REGISTRIES: SECURE ACCESS TO IMAGES

Image governance & private registries

- Are there access controls on the registry? How strong are they?
- What security meta-data is available for your images?
- How is the data kept up-to-date?



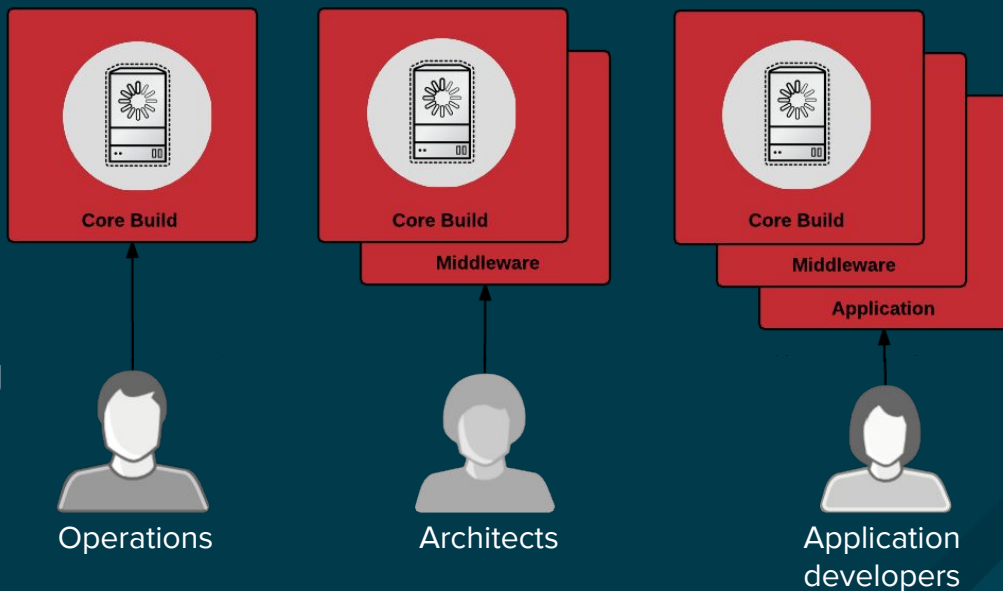
- Red Hat Container Registry
- Policies to control who can deploy which containers
- Certification Catalog
- Trusted content with security updates



4 MANAGING CONTAINER BUILDS

Security & continuous integration

- Layered packaging model supports separation of concerns
- Integrate security testing into your build / CI process
- Use automated policies to flag builds with issues
- Ensure builds always use the latest base image
- Trigger automated CI process



5

MANAGING CONTAINER DEPLOYMENT

Security & continuous deployment

- Monitor image registry to automatically replace affected images
- Use policies to gate what can be deployed: e.g. if a container requires root access, prevent deployment
- Monitor application health & behavior

```
$ oc describe scc restricted
Name:                restricted
Priority:             <none>
Access:
  Users:             <none>
  Groups:            system:authenticated
Settings:
  Allow Privileged:  false
  Default Add Capabilities: <none>
  Required Drop Capabilities: KILL,MKNOD,SYS_CHROOT,SETUID,SETGID
  Allowed Capabilities: <none>
  Allowed Volume Types: configMap,downwardAPI,emptyDir,persistentVolumeClaim,secret
  Allow Host Network: false
  Allow Host Ports:   false
  Allow Host PID:     false
  Allow Host IPC:     false
  Read Only Root Filesystem: false
  Run As User Strategy: MustRunAsRange
    UID: <none>
    UID Range Min: <none>
    UID Range Max: <none>
  SELinux Context Strategy: MustRunAs
    User: <none>
    Role: <none>
    Type: <none>
    Level: <none>
  FSGroup Strategy: MustRunAs
    Ranges: <none>
  Supplemental Groups Strategy: RunAsAny
    Ranges: <none>
```

6

CONTAINER ORCHESTRATION & SECURITY

CONTAINER ORCHESTRATION & CLUSTER MANAGEMENT
(KUBERNETES)

NETWORKING

STORAGE

REGISTRY

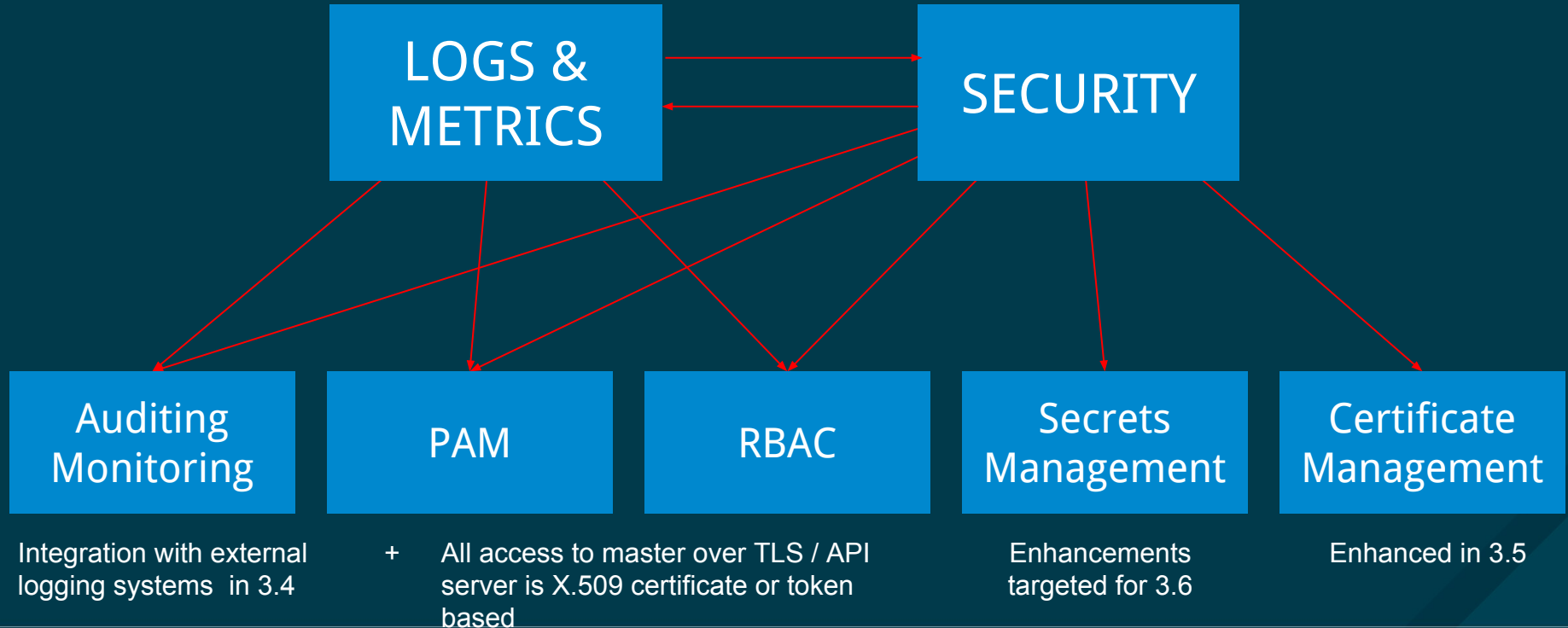
LOGS &
METRICS

SECURITY

INFRASTRUCTURE AUTOMATION & COCKPIT

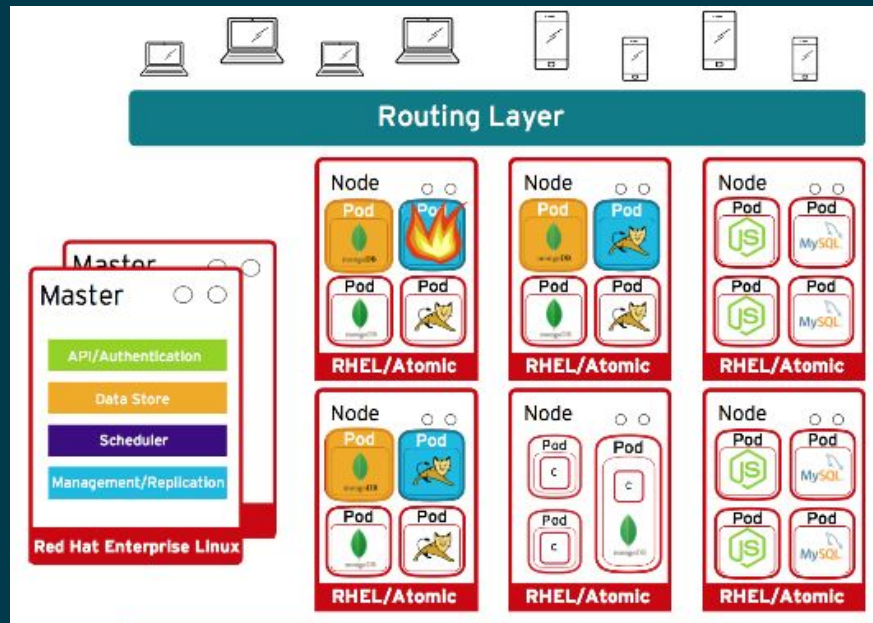
6

CONTAINER ORCHESTRATION & SECURITY



7 CONTAINER MULTITENANCY & NETWORK DEFENSE

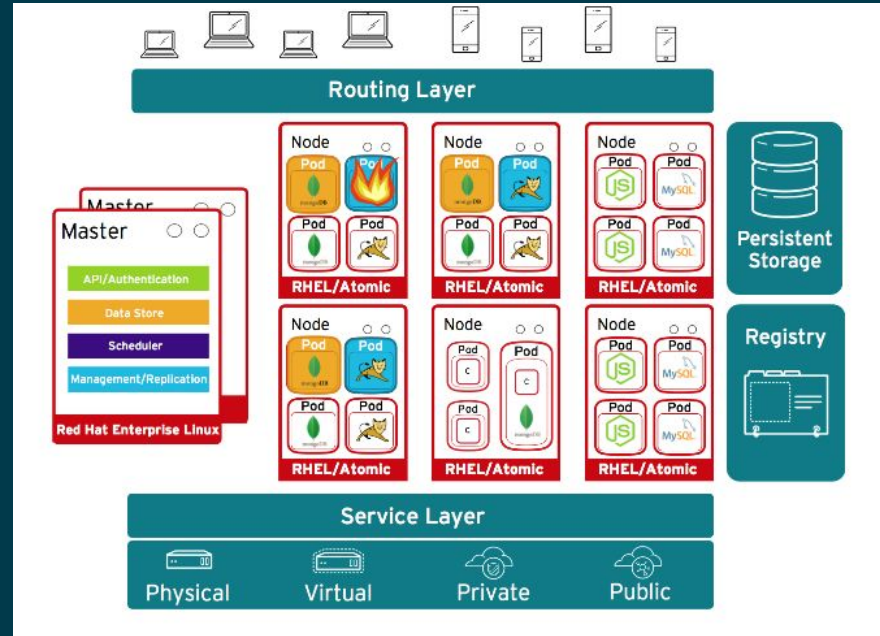
- Segment traffic to isolate users, teams, applications within a single cluster
- Manage egress traffic to meet existing firewall policies
- Tech-preview network policy plug-in allows isolation policies to be configured for individual pods



8 ATTACHED STORAGE

Secure storage by using

- SELinux access controls
- Secure mounts
- Supplemental group IDs for shared storage



9 API MANAGEMENT

Container platform & application APIs

- Authentication and authorization
- LDAP integration
- End-point access controls
- Rate limiting

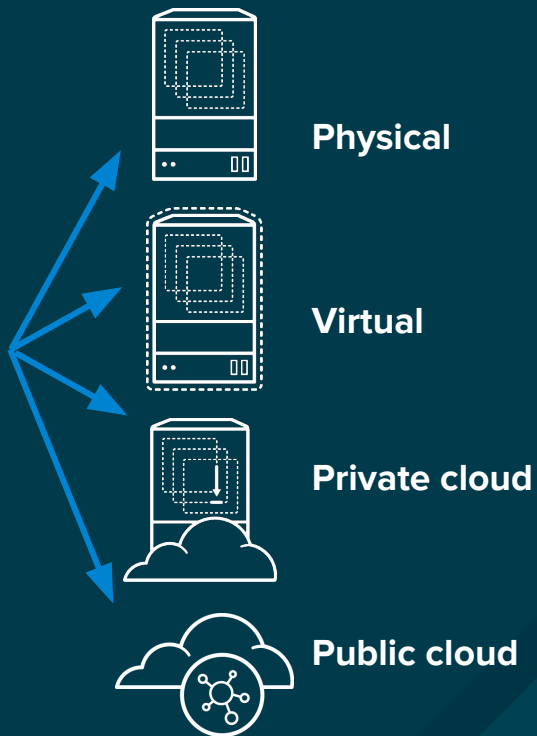
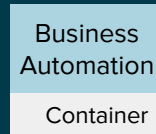
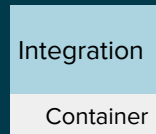
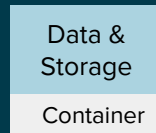
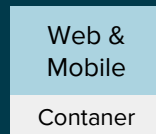
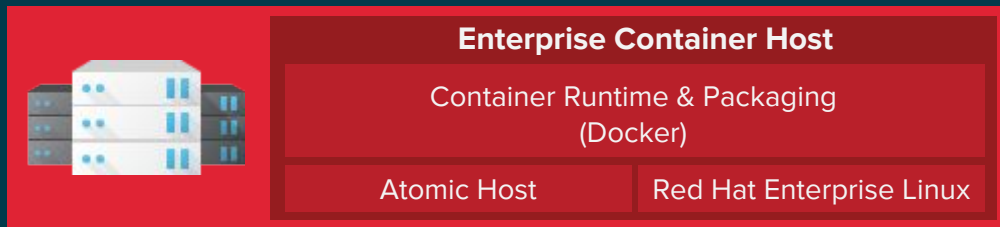
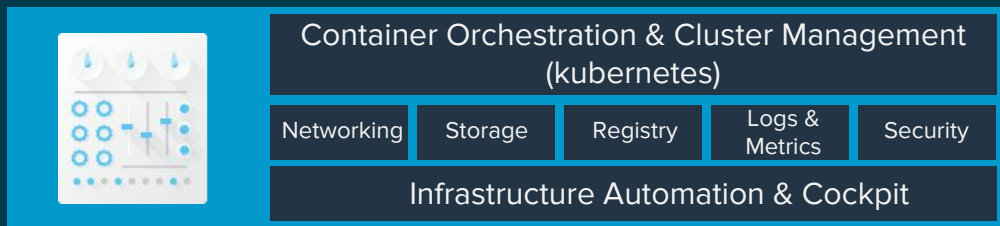
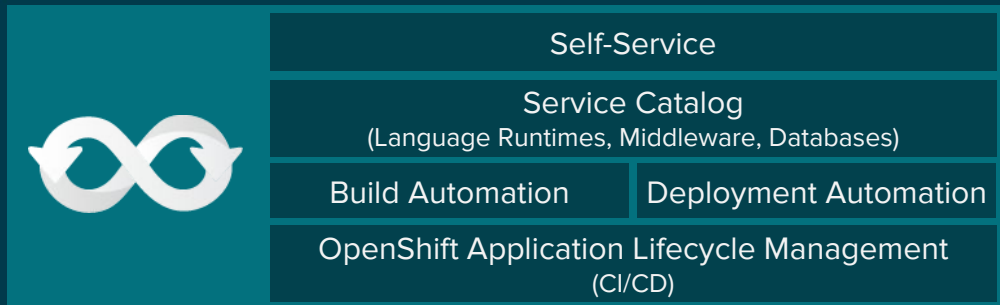


FUTURE: FEDERATED CLUSTERS ROLES & ACCESS MANAGEMENT

Securing federated clusters
across data centers or
environments

- Authentication and authorization
- API endpoints
- Secrets
- Namespaces





READ THE WHITEPAPER

Ten Layers of Container Security

RED HAT
SUMMIT

THANK YOU



plus.google.com/+RedHat



facebook.com/redhatinc



linkedin.com/company/red-hat



twitter.com/RedHatNews



youtube.com/user/RedHatVideos

The logo consists of a red speech bubble shape pointing downwards, containing the text "RED HAT" in a smaller font above "SUMMIT" in a larger, bold font.

RED HAT
SUMMIT

LEARN. NETWORK.
EXPERIENCE
OPEN SOURCE.