

RED HAT  
**SUMMIT**

# DevSecOps the open source way

Gordon Haff, Technology Evangelist Red Hat @ghaff

William Henry, Senior Consultant, DevOps Strategy Red Hat @ipbabble

3 May 2017

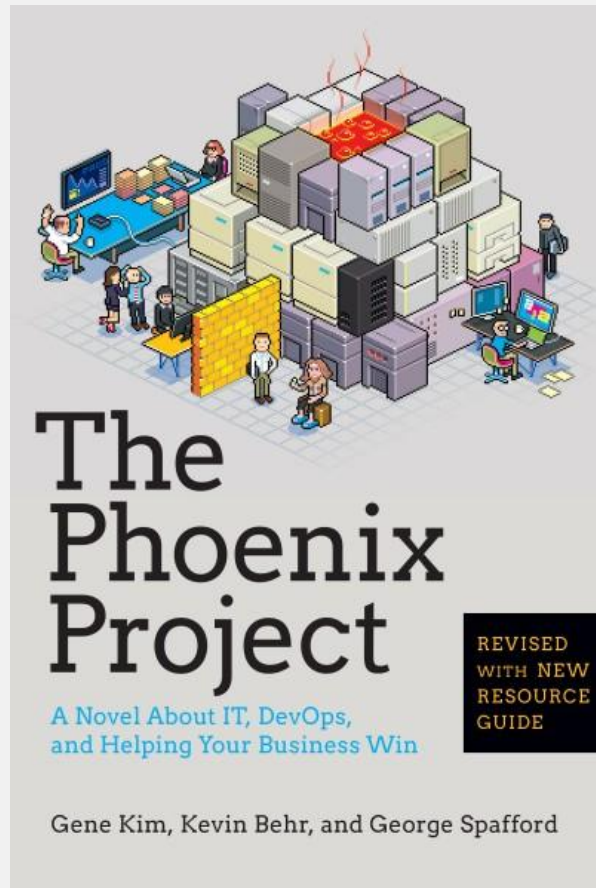
# What is DevSecOps?

# HOW DEVS AND OPS VIEW SECURITY



# WHY DevSecOps?

- DevSecOps practitioners say it's about how to **continuously integrate** and **automate** security at **scale**
- DevOps “purists” point out that security was always part of DevOps
- Did people just not read the book? Are practitioners skipping security?





# GLASS HALF EMPTY, GLASS HALF FULL

*“... we estimate that fewer than 20% of enterprise security architects have engaged with their DevOps initiatives to actively and systematically incorporate information security into their DevOps initiatives; and fewer still have achieved the high degrees of security automation required to qualify as DevSecOps.”*

*“By 2019, more than 70% of enterprise DevOps initiatives will have incorporated automated security vulnerability and configuration scanning for open source components and commercial packages, up from less than 10% in 2016.”*

# Characteristics of the new paradigm

# THE WORLD IS CHANGING

THEN	NOW
IT as a supporting cost center	Technology driving new revenue
Established industry structures	“Software is eating the world”
Ad hoc decision making	Data-driven real-time response & analytics
Multi-year product cycles	Rapid iterative service refreshes
Focus on individual product success	Achieve ecosystem scale



# REUSE

- Modular apps
- Open source repos

# MICROSERVICES

- Single-function components
- Bounded context
- Two-pizza teams
- RESTful interfaces

# AUTOMATION

- Automate (most of the things)
- Standardization
- Repeatability
- Have you done it more than once?

# IMMUTABILITY

- Restart instead of repair
- Know the state

# PERVASIVE ACCESS

- No firewalls
- Access through APIs
- Reactive design

# SPEED

- Fast to develop
- Fast to deploy
- Rapid iteration

# SOFTWARE-DEFINED

- Flexible
- Scalable
- Portable

# FLEXIBLE DEPLOYS

- Blue-green
- Canary



# CONTAINERS

- Self-describing software
- App stores
- Portable workloads

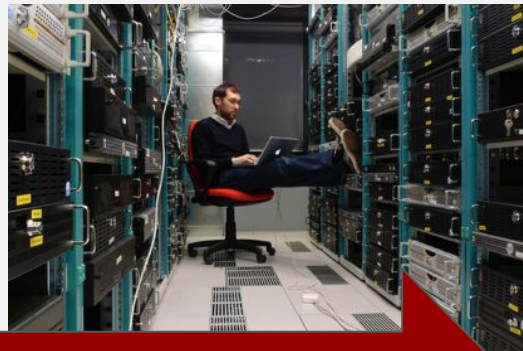
# RAPID TECH CHURN

- Open source innovation
- Loosely-coupled projects
- New ecosystems



Dev

Ops



Microservices

Automation

Immutability

Reuse

Pervasive access

Flexible deploys

Speed

Containers

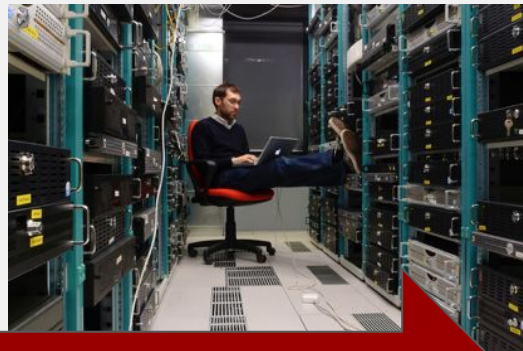
Rapid tech churn

Software-defined



**Dev**

**Ops**



Microservices

Automation

Immutability

Reuse

Pervasive access

Flexible deploys

Speed

Containers

Rapid tech churn

Software-defined

# DevSecOps: The Red Hat open source way

# YOU MANAGE RISK BY - perhaps hide this slide

- Securing the Assets
- Securing the Dev
- Securing the Ops
- Securing the APIs

# SECURING THE ASSETS

- Building code
  - Watching for changes in how things get built
  - Signing the builds
- Built assets
  - Scripts, binaries, packages (RPMs), containers (OCI images), machine images (ISOs, etc.)
  - Registries (Service, Container, App)
  - Repositories (Local images and assets)



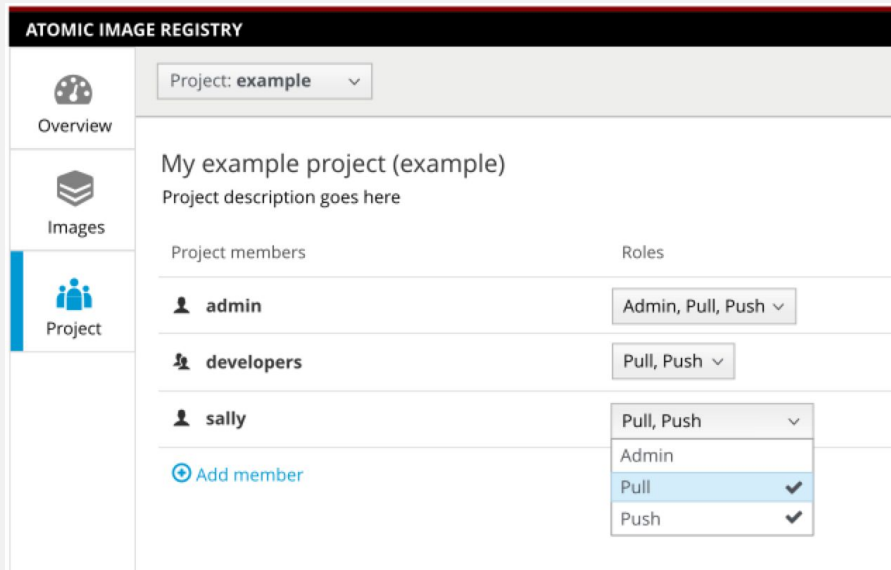
Safe at Titan Missile Museum

[https://upload.wikimedia.org/wikipedia/commons/5/59/Red\\_Safe%2C\\_Titan\\_Missile\\_Museum.jpg](https://upload.wikimedia.org/wikipedia/commons/5/59/Red_Safe%2C_Titan_Missile_Museum.jpg)

# SECURING THE SOFTWARE ASSETS - E.G. IMAGE REGISTRY

## Public and private registries

- Do you require a private registry?
- What security meta-data is available for your images?
- Are the images in the registry updated regularly?
- Are there access controls on the registry? How strong are they? Who can push images to the registry?



The screenshot displays the 'ATOMIC IMAGE REGISTRY' interface. On the left, a sidebar contains navigation options: 'Overview' (selected), 'Images', and 'Project'. The main content area shows the configuration for a project named 'example'. It includes a 'Project: example' dropdown, a description field, and a table of project members with their roles. A dropdown menu is open for the 'sally' member, showing options for 'Admin', 'Pull', and 'Push', with 'Pull' and 'Push' selected.

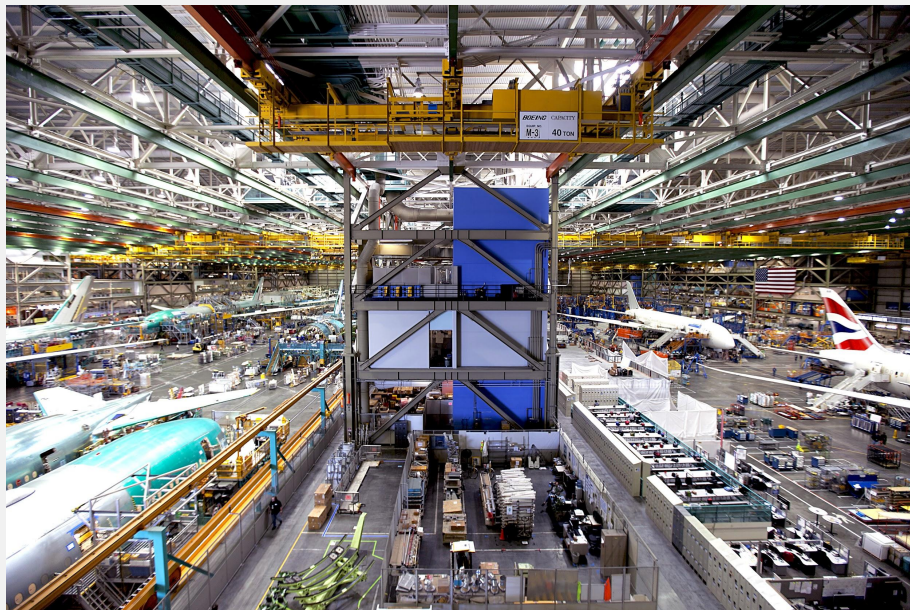
Project members	Roles
admin	Admin, Pull, Push
developers	Pull, Push
sally	Pull, Push

[Add member](#)



# SECURING THE DEVELOPMENT PROCESS

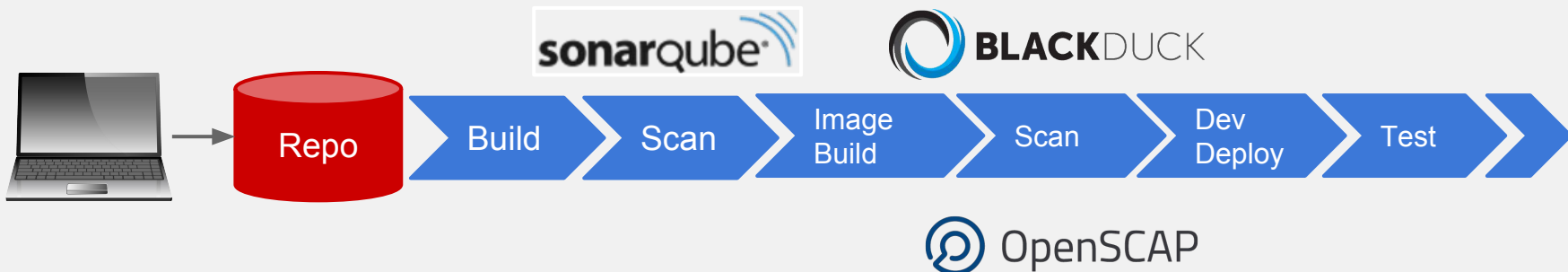
- Potentially lots of parallel builds
- Source code
  - Where is it coming from?
  - Who is it coming from?
- Supply Chain Tooling
  - CI tools (e.g. Jenkins)
  - Testing tools
  - Security Tools (e.g. Black Duck, Sonatype)



Boeing's Everett factory near Seattle

[https://upload.wikimedia.org/wikipedia/commons/c/c8/At\\_Boeing%27s\\_Everett\\_factory\\_near\\_Seattle\\_%289130160595%29.jpg](https://upload.wikimedia.org/wikipedia/commons/c/c8/At_Boeing%27s_Everett_factory_near_Seattle_%289130160595%29.jpg)  
Creative Commons

# SECURING THE DEVELOPMENT



Ensure the application code is compliant.  
Ensure the pipeline is not compromised.

# SECURING THE OPERATIONS

- Deployment
  - Trusted registries and repos
  - Signature authenticating and authorizing
  - Image scanning
  - Policies
  - Ongoing assessment with automated remediation
- Lifecycle
  - Blue Green and A/B continuous deployments
  - Monitoring deployments
  - Possibly multiple environments
  - Multiple threats



Mission Control - Apollo 13

[https://cl.staticflickr.com/4/3717/9460197822\\_9f6ab3f30c\\_b.jpg](https://cl.staticflickr.com/4/3717/9460197822_9f6ab3f30c_b.jpg)



# SECURING THE OPERATIONS - FRESHNESS

- Freshness Grade for container security.
- 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



# SECURING THE OPERATIONS - LOGGING

## EFK Stack

- Elasticsearch, Fluentd, Kibana
- Based on log aggregation
- Event system - all events container, system, kubernetes, captured by EFK and issues or errors
- Good for ad hoc analytics
- Good for post mortem forensics because of extensive log information



# SECURING THE OPERATIONS - METRICS

Metrics tools tend to make more use of APIs than logs. You need to figure out your organizational needs.

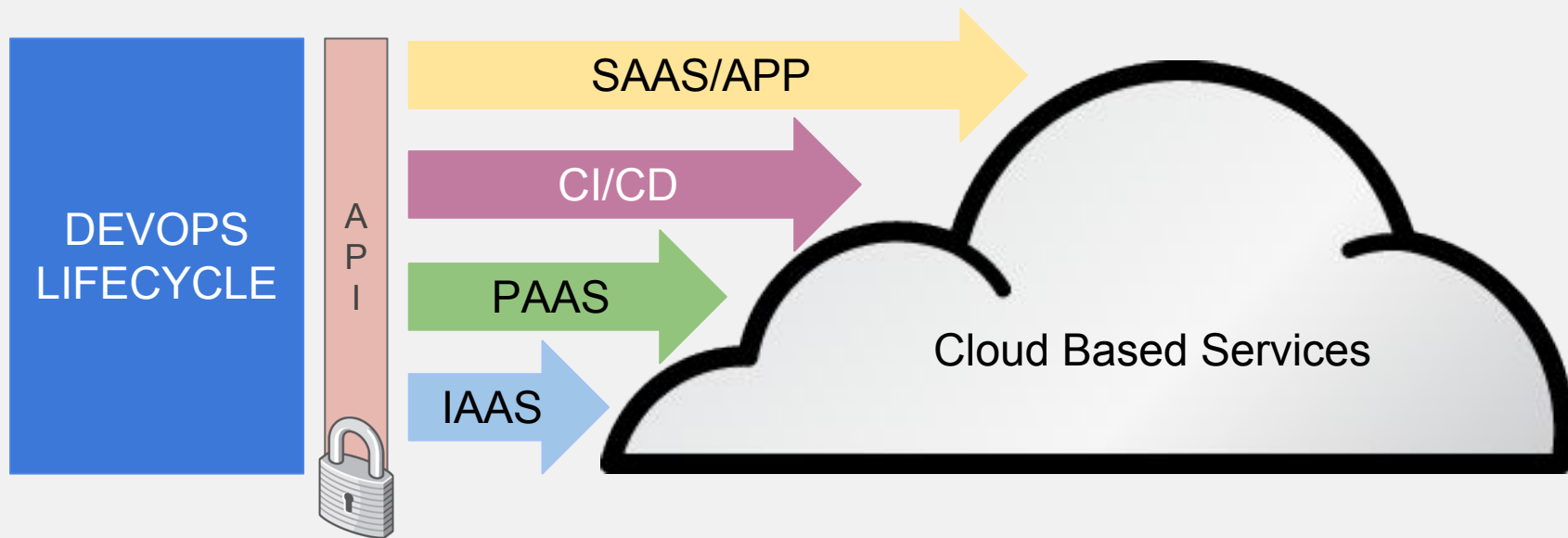


- Ideal for large scale central IT teams with lots of apps.
- OpenShift ships with Hawkular



- Prometheus is ideal for WebScale DevSecOps

# SECURING THE APIS



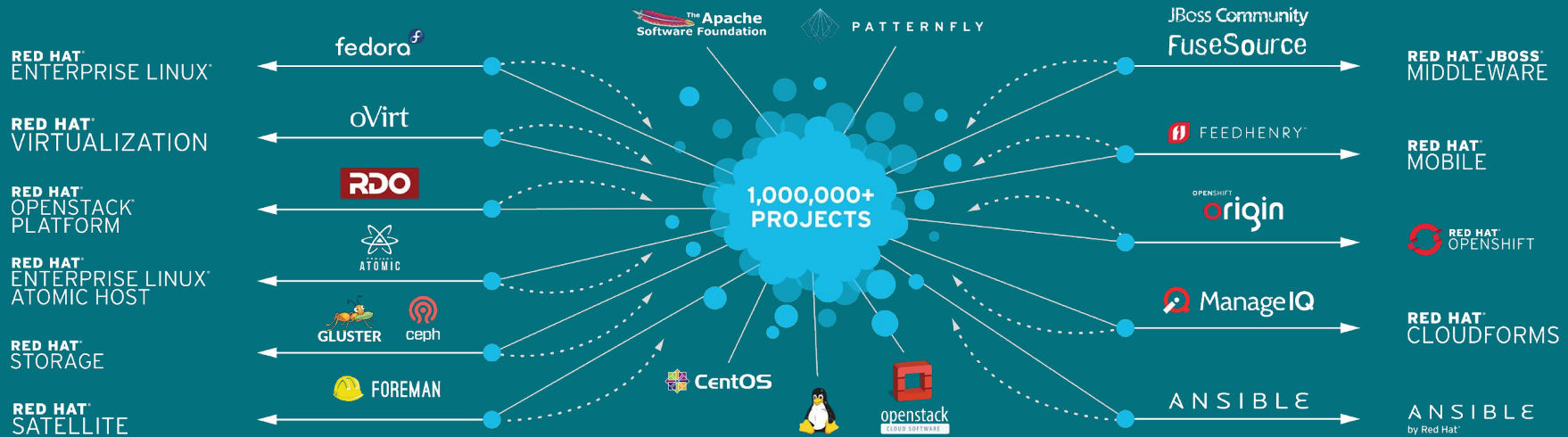
Modern Architectures are API driven requiring a DevOps approach to API management. Visibility, routing, and authorization are key security concerns.

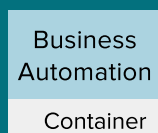
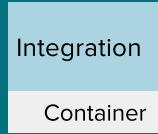
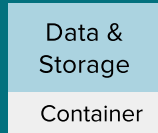
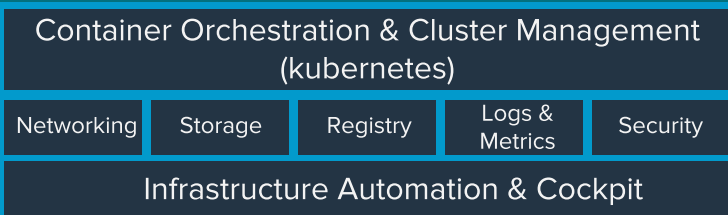
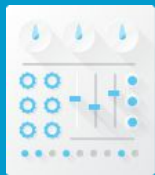
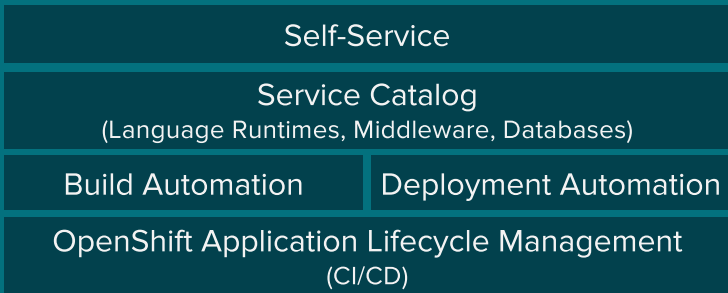


# MANAGING OR TRANSFERRING RISK

How can a CIO feel confident that their organization is managing all the valuable Open Source?

If you are trying to manage your risk then where and how you get your open source matters.





**Physical**



**Virtual**



**Private cloud**



**Public cloud**



RED HAT  
**SUMMIT**

# THANK YOU



[plus.google.com/+RedHat](https://plus.google.com/+RedHat)



[facebook.com/redhatinc](https://facebook.com/redhatinc)



[linkedin.com/company/red-hat](https://linkedin.com/company/red-hat)



[twitter.com/RedHatNews](https://twitter.com/RedHatNews)



[youtube.com/user/RedHatVideos](https://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, both in white.

**RED HAT**  
**SUMMIT**

LEARN. NETWORK.  
EXPERIENCE  
OPEN SOURCE.