RED HAT SUMMIT

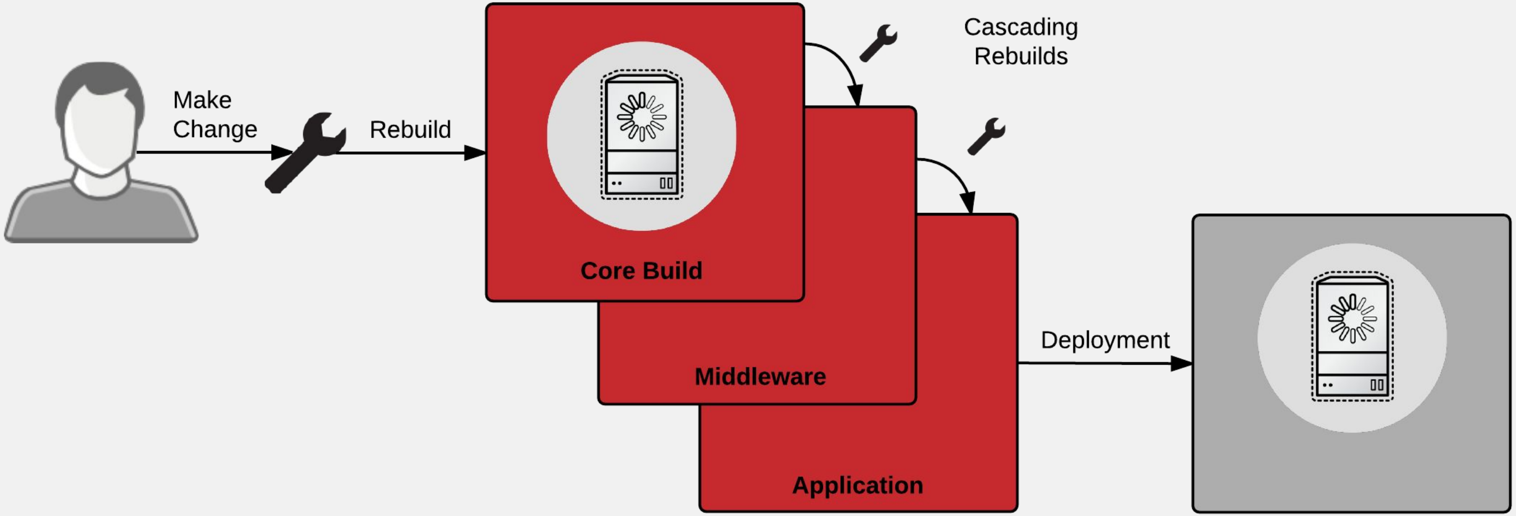
Secure your enterprise software supply chain with containers

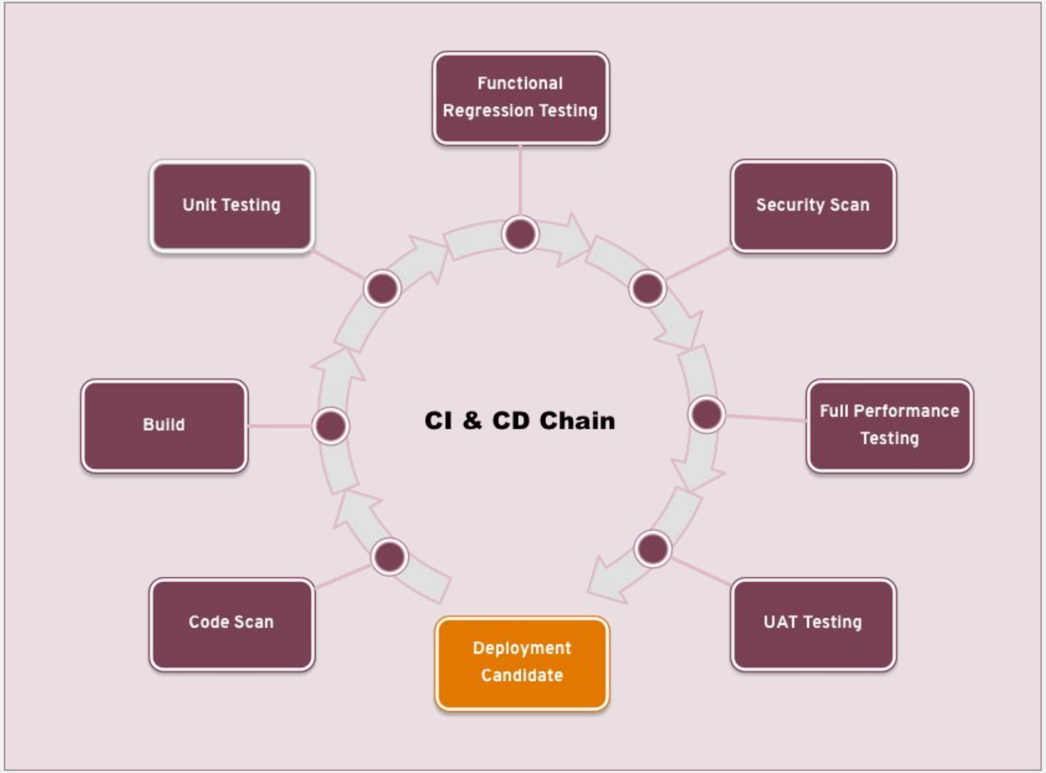
Curtis Yanko (@onCommit)
Sr. Principal Architect, Sonatype

Randy Kilmon (@randykilmon)
VP of Engineering, Black Duck

Zohaib Khan (@zeebluejay)
Principal Architect, Red Hat

Scott McCarty (@fatherlinux)
Sr. Principal Technical Product Marketing Manager, Red Hat


Make
Change
Rebuild
Core Build
Cascading
Rebuilds
Middleware
Application
Deployment


Functional Regression Testing
Security Scan
Full Performance Testing
UAT Testing
Deployment Candidate
Code Scan
Build
Unit Testing
CI & CD Chain

General (8 Mins)

1. What percentage of your customers use Docker in production? What are the main concerns about moving Docker to production? Will the containerized model dominate? (All)
2. What are the basic security considerations for containers and why do they matter? (All)

Security (9 Mins)

1. What are the elements of a secure software supply chain and how to get them right the first time? (Zohaib)
2. A container running in production is identified as having a vulnerable CVE. Given that most containers would be short lived anyway, why do we need to be so vigilant about identifying and patching them? (Randy)
3. How do you do CVE patching in a containerized world? (Scott)
4. How do you audit/verify what, where (provenance) and when (over a time series) content is being used in a container? Which components should be tracked? (Scott, Randy, Curtis)

Quality & Agility (8 Mins)

1. If you have a lot of containerized applications, how do you manage container content efficiently at scale? How do you manage the relationship between dev and ops. (Soott)
2. How does building a quality software supply chain as part of continuous delivery actually help reduce initial shipping time as well as mean time to identify and remediate issues in production? (Zohaib, Curtis)
3. Understanding the security profile of a container is vital, but what other aspects of the container lifecycle are exploitable and what precautions can be taken? (Curtis)

Tools (8 Mins)

1. What types of tools can improve governance, monitoring, inventory and auditing of components that are used inside of containers? (Zohaib)
2. If a component or container is discovered to have a defect, what tools can accelerate mean time to identify and remediate the issue? (Randy, Curtis)
3. Provenance and trust are critical in today's highly agile and security sensitive landscape, are there tools that track provenance and maintain trust of containers and the components inside of containers? (Scott)

Organizational Change (11 Mins)

1. Why should people care about the provenance of components inside of Docker containers? (Zohaib)
2. Since a container essentially embeds developer controlled content (e.g. Ruby Modules) and operations controlled content (Linux User Space), who has authority to bring which components inside the company from outside sources? (Randy, Curtis, Scott)
3. How can we control which 3rd party and open source components our teams are allowed to use? (Randy, Curtis)

RED HAT SUMMIT

LEARN. NETWORK.
EXPERIENCE OPEN SOURCE.

Further Reading & Citations

1. Before You Initiate a Docker Pull: http://red.ht/28VD8yU
2. How to Run a More Secure Non-Root User Container: http://bit.ly/28YI7hS
3. Building a Secure and Manageable Container Software Supply Chain: http://red.ht/28VDzt1
4. Meeting the CIS Docker Benchmark with RHEL7 and RHEL Atomic: http://bit.ly/28VjWNE
5. Docker's New Security Advisories and Untrusted Images: http://bit.ly/28ULELZ
6. Black Duck at Red Hat Summit: http://bit.ly/2968fpC
7. Free Docker Regsitry from Sonatype: http://bit.ly/29c1nY3
8. Rugged DevOps and Software Supply Chain White Papers: http://bit.ly/290uMpG