Ongoing Authorization In Practice

October 20, 2016
Leo Garciga – JIDO J6/CTO
Ryan Skousen – Chief Engineer, Booz Allen Hamilton
Agenda

• Who is JIDO and what’s our mission?
• Secure Agile at JIDO – The Foundation
• DevOps is Awesome!
• What about Security/Accreditation?
• Bringing it all together
• Questions???
JIDO’s Mission

• JIDO is a QRC: Quick Reaction Capability
  – Bring timely solutions to warfighters
  – Focused on “what’s real” (0-2 yrs out)
  – “Operationalize” new technologies
  – Defeat Threat Networks

• JIDO J6 Mission IT
  – Built a Big Data analytic platform, “Catapult”, and tool suite based on real-time, tactical needs
  – Embed with users world-wide to understand data available, analytic methodologies, & capability/data gaps
  – Solutions required same day at times

“Adapt or Die”
Secure Agile at JIDO – The Foundation

• Running “true” Agile SDLC for 5 years
• Sprints: 3 weeks, release to production (almost) every sprint
• Approved Catapult as a “system of systems” each with a “type authorization”
  – Design ATO concurrently with technical design
  – Secure “docking port” for new tools
  – “Widgets” or webapps get an IT Product assessment only
• Continuous Integration already implemented with nightly security scans
• Release management with traditional CM/CCB still hard
  – 50+ releases per year (historically)
  – JIDO able due to IA/Ops/Dev all report to one boss

*Agile alone is not enough!*
• Started JIDO DevOps evolution in 2015
• Adopted tooling to support a DevOps pipeline:
  – Linux Containers (Docker, Trusted Registry, etc.)
  – Secure dependency management (SonaType)
  – Real-time container security (Twistlock – awesome)
• Security & compliance built in up-front
• Goal – completely automated deployments from code to production
  – SMALL changes – like every commit
  – NO manual/human review gates
  – Adoptable by other agencies

Secure DevOps, anyone?
What About Security/Accreditation?

• Doesn’t DevOps bypass many of the security checks?
• We believe the vision and intent of policy supports an automated delivery pipeline (NIST SP 800-37):

  – Appendix F.4 - Ongoing Authorization: "The ultimate objective is to achieve a state of ongoing authorization where the authorizing official maintains sufficient knowledge of the current security state of the information system … to determine whether continued operation is acceptable based on ongoing risk determinations… Formal reauthorization actions are avoided in situations where the continuous monitoring process provides authorizing officials the necessary information to manage the potential risk…"

  – Appendix G - Continuous Monitoring: “… The monitoring program is integrated into the organization’s system development life cycle processes… Near real-time risk management of information systems can be facilitated by employing automated support tools to execute various steps in the RMF including authorization-related activities.”

Ongoing Authorization is Secure Agile + DevOps + Continuous Monitoring
JIDO Ongoing Authorization Pipeline

- Container-compatible platform certified & approved up front
- OS image and common containers (e.g. Java) are hardened & approved up front, registered in Docker Trusted Registry
- Workflow is integrated with Jenkins, Git, SonarQube, etc. for a complete picture
- Features/changes are marked in issue system as requiring manual CCB review or not before coding begins. If CCB flag is set, deployment stops at pre-production until a final concurrence is marked in JIRA

**Automated, Secure Delivery Reduces Risk**
Major Take-Aways

• Secure design and planning throughout SDLC
• Containers for standardized deployment packaging
• Secured, transparent DevOps pipeline.
  – Prohibits tampering, provides monitoring, and traceability.
  – Escalation based on code triggers (code delta, coverage)
• Type-accredited platform to receive and run containers
• It’s like having a trusted candy factory, packaging goodies into bulletproof briefcases, transporting through a point-to-point hyperloop, delivering to candy shops with turrets – Really need to lick every lollipop?
Questions?
Security Checks Performed

- IA Design validation (daily DevOps scrum before coding begins)
- Operating System (host systems monthly, base Docker images on creation)
  - STIGs/CIS Benchmark
  - SCAP scanning
- Dependency library scanning (ongoing)
- Static Code Analysis (on build)
- Unit Testing (on build)
- Automated UI testing (on build)
- Active Code Analysis/Web penetration testing (on build, ongoing on production)
- Image Integrity Checking (Docker Trusted Registry - on build & deploy)
- Approved Image validation (on build & deploy)
- Container Inspection & Scanning (Twistlock - on deploy & ongoing)
  - CIS benchmarks
  - OWASP top 10
  - Vulnerable library identification
- Port/vulnerability scanning (ongoing)
- HBSS on hosts (ongoing)