DEVSECOPS ACADEMY
TRANSFORMING DOD’S WORKFORCE:
WINNING THE FIGHT WITH DEVSECOPS
AND DIGITAL INNOVATION

“WINNING THE FIGHT ANYWHERE – DEMANDS SOFTWARE EVERYWHERE.”
SEAN BRADY | LEARNING DIRECTOR, SOFTWARE ACQUISITION

May 9, 2019. Send questions to: Sean.Brady@dau.mil; Defense Acquisition University.
OBJECTIVES

The Why -- Urgency for Change & DevSecOps

- DoD SW Acquisition State of Play and Challenge
- What’s Causing this?
- Congressional Software & DevSecOps Initiatives
- Innovation: Winning on the Modern Battlefield

What is DevSecOps?

- What is Agile? DevOps? Shift Left?
- Where’s the Sec in DevOps?
- DoD Enterprise Platform
- What Silos Must be Broken?

The How: Case Study -- DevSecOps Success & Value in DoD

DevSecOps Academy Concept

- Growth Mindset: Implementation Skills and Culture
- Partnership and Community of Practice

Theme: “Integrating Agile into Government”
PROBLEM STATEMENT AND DOD’S CHALLENGE

“How do we transform the workforce, practices, competencies, training and culture within the world’s largest engineering organization & bureaucracy (200K+ professionals)...in order to radically accelerate the adoption of modern software development practices at unprecedented-scale, across the entire DoD?”

James Mattis, Former Secretary of Defense, established “business reform” as one of three priorities for “protecting our people and ensuring the survival of our freedoms.”

This reform is impossible without confronting the software challenge. “U.S. Troops Should Not Be Sent Into Fair Fights”

General Joseph Dunford, Chairman, Joint Chiefs of Staff
“Our adversaries are acquiring capabilities not previously anticipated and are doing so at a pace that now challenges U.S. technological superiority.

The US must have the ability to quickly respond to adversary advancements & update our systems accordingly.

Rapid & continuous SW development will be essential to achieving this outcome.

SW development in the commercial world has undergone significant change in the last 15 years, while DoD has continued to use techniques developed in the 1970s - 1990s.

The Department must change.”

-2018 DSB on Software

<table>
<thead>
<tr>
<th>Most Important to Warfighter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed of Investment to Meet Needs</td>
</tr>
<tr>
<td>Time-to-Market/Warfighter</td>
</tr>
<tr>
<td>Responsiveness to Rapidly Changing Threats: Adaptability of Systems</td>
</tr>
<tr>
<td>Rapid Adoption of Emerging SW Tech (Force Multipliers)</td>
</tr>
<tr>
<td>Innovation (tech superiority)</td>
</tr>
<tr>
<td>Risk Aversion</td>
</tr>
<tr>
<td>Detailed Upfront Plans and Specifications</td>
</tr>
<tr>
<td>Bureaucratic Compliance</td>
</tr>
</tbody>
</table>

- Agile in Government Conference | May 9, 2019
Our adversaries are acquiring capabilities not previously anticipated and are doing so at a pace that now challenges U.S. technological superiority.

The US must have the ability to quickly respond to adversary advancements & update our systems accordingly.

Rapid & continuous SW development will be essential to achieving this outcome.

SW development in the commercial world has undergone significant change in the last 15 years, while DoD has continued to use techniques developed in the 1970s - 1990s.

The Department must change.”

-2018 DSB on Software
“As we reorganize the way we do business the thread that runs through all of our programs and all that we do is software and I believe that we need to catch up with the private sector and make sure we are using contemporary software development processes,”

The Honorable Ellen Lord,
Under Secretary of Defense,
Acquisition and Sustainment

This attribute map compares the key factors in Department of Defense (DOD)-sponsored and Privately held companies that participate in the Government Acquisition Process. *Rankings based off effectiveness of attribute with ‘1’ being least conducive to the acquisition process benefiting product to end user and ‘5’ being most beneficial.
The Honorable Ellen Lord, Under Secretary of Defense, Acquisition and Sustainment

As we reorganize the way we do business the thread that runs through all of our programs and all that we do is software and I believe that we need to catch up with the private sector and make sure we are using contemporary software development processes,”

This attribute map compares the key factors in Department of Defense (DOD)-sponsored and Privately held companies that participate in the Government Acquisition Process. *Rankings based off effectiveness of attribute with '1' being least conducive to the acquisition process benefiting product to end user and ‘5’ being most beneficial.
“Software is assessed among the most frequent and most critical challenges, driving program risk on ~60% of acquisition programs.”

Source: Defense Science Board, February 2018 Report
• Certain application domains (e.g., C4ISR) have an **obvious affinity for rapid deployment cycles** (e.g. cloud-native apps)
• We also must confront **unique needs of weapon system developers** – the **largest portion of the major acquisition portfolio**

• SW is subsystem with cyber-physical dependencies/interfaces (e.g., fighter jets, ballistic missiles, radars)
• may be **inherent limit to the fidelity of integration labs** or the **types of continuous end-to-end testing** possible (e.g. flight test; multi-sensor, multi-jet fights; cost-prohibitive live-fire testing).
what’s causing this? manual build/test/release – key threat to mission

...along with Late Integration and Defect Discovery
DoD has encountered significant late-stage, “big bang” integration characterized by
• long cycle times between deliveries (18-24 months)
• manual, error-prone build times measured in weeks, months, or even years

Lack of modern software development practices on today’s complex systems has resulted in integration nightmares.

For example:
• late stage “big bang” integration with manual testing for 5 or more years
• multiple development streams occurring in configuration item (CI) silos
• manual configuration management (CM) and platform configuration
• varying development, integration and production environments
• exceedingly long build cycles and build times (6-12 months to manually deploy and test code in production environment)
• code and unit test cycles occurring in weeks as opposed to hours
• end-to-end, system integration testing occurring in months as opposed to days or hours
Most companies deploy weekly, daily, or hourly. That was then. This is now.

Gaining first-mover advantage

DIGITAL NATIVES? CONGRESSIONAL MANDATES

FY2018 NDAA

• Sec. 872: Defense Innovation Board analysis of software acquisition regulations.
• Sec 873: Pilot program to use agile or iterative development methods on major programs.
  – One software intensive warfighting system per service and one defense-wide
  – Two to eight Defense Business Systems
• Sec 874: Identify 4-8 SW development activities as pilot programs to use agile methods
• Sec 891: DAU training in support of sections 873/874
  – Mandatory for those involved in Sec 873/874 programs / Offered to other programs by request

FY2019 NDAA

  – DSB Report Feb 2018 - seven recommendations
• Sec 869: Implementation of Pilot Program to Use Agile or Iterative Development Methods Requested Under Section 873 of NDAA FY2018
  – Additions to FY18 Sec 873 list -- Community of Practice advising on agile or iterative development
DSB calls for SW factory / DevSecOps / innovation (Lean Startup & Design Thinking) competency


Not later than 18 months after the date of the enactment of this Act, the Secretary of Defense shall, except as provided under subsection (b), commence implementation of each recommendation submitted as part of the final report.

1. **Software Factory** – A key evaluation criteria in the source selection process should be efficacy of the offeror’s software factory.
   • DoD has limited iterative development expertise – focus on acquisition

2. **Continuous Iterative Development** – DoD and defense industrial base partners should adopt continuous iterative development best practices.
   • identify approaches and deliver minimum viable product (MVP)
   • establish MVP in its formal acquisition strategy, and arrange for the warfighter to adopt the IOC as an MVP for evaluation and feedback;
   • ... require all programs entering Milestone B to implement these iterative processes for Acquisition Category (ACAT) I, II, and III programs.

5. **Workforce** –
   • Services need to develop workforce competency (prioritize acquisition strategy, source selection)
   • DAU develop curricula to develop SW-informed PMs, sustainers, acquisition specialists
DIB report to Congress fast approaching (May ’19)

SOFTWARE ACQUISITION & PRACTICES (SWAP) - WORKFORCE SUBGROUP / FY18 NDAA - §872

streamlining and improving the efficiency and effectiveness of software acquisition in order to maintain defense technology advantage;

TL; DR Appendix B. SWAP Report Appendix F. SWAP Working Group Reports (DIB remix)

- Acquisition Strategy
- Appropriations
- Contracts
- Data and Metrics
- Infrastructure
- Modernization/ Sustainment
- Requirements
- Security Certification/Accreditation
- Testing and Evaluation
- Workforce
DIB report to Congress fast approaching (May ’19)

March Report

- **Path Forward 1:** Most importantly, Gov and industry must come together to implement a DevSecOps culture and approach to SW, as used in industry.

- Make use of existing authorities such as OTAs and mid-tier acquisition (Sec 804) to implement a DevSecOps approach to acquisition to greatest extent possible

- Special Experience Identifier for mil & civ acquisition professionals indicating they have necessary experience and training to serve on a software acquisition team. ...mandatory requirement to lead any SW procurement

- **A1:** USD(A&S), with SAEs, select programs using DevSecOps to utilize new SW acquisition pathway; develop and implement training at DAU on new SW pathway for all acquisition communities (FM, PM, IT, SE, etc.)

- C2: Leverage existing training; add modern software development content
- C2: Create SW continuing education programs and requirements for CIOs, SAEs, PEOs and PMs
- C3: Create & provide training opportunities via boot camps & rotations to gain hands-on DevSecOps experience
CONFRONTING THE CHALLENGE & URGENCY

• What is DevSecOps?
  • The software integrated tools, services, and standards that enable partners and programs to develop, deploy, and operate applications in a secure, flexible and interoperable fashion.

• Why should I care?
  • Software and cybersecurity pervades all aspects of DoD's mission (from business systems to weapons systems to Artificial Intelligence to cybersecurity to space) - establishing DevSecOps capabilities will:
    • Deliver applications rapidly and in a secure manner, increasing the warfighters competitive advantage
    • Bake-in and enforce cybersecurity functions and policy from inception through operations
    • Enhance enterprise visibility of development activities and reduce accreditation timelines
    • Ensure seamless application portability across enterprise, Cloud and disconnected, intermittent and classified environments
    • Drive DoD transformation to Agile and Lean Software Development and Delivery
  • Leveraging industry acquisition best practices combined with centralized contract vehicle for DevSecOps tools and services will enable rapid prototyping, real-time deployments and scalability
  • We cannot be left behind: China, Russia and North Korea are already massively implementing DevOps

Nicolas Chaillan - DoD Enterprise DevSecOps Platform (Software Factory) Initiative
Dr. Will Roper, Assistant Secretary of the Air Force for Acquisition, Technology and Logistics: “Software intensive programs are almost all over cost, over schedule.”

These delays come at an increasingly severe price to our Warfighters on the battlefield.

On a modern battlefield and in a future war, Dr. Roper goes on to say “we could be changing software every day as a necessary factor for winning.”
INNOVATION & LEARNING

The Smart Machine Age (SMA)

• UVA Darden Ed Hess: **NO. 1 JOB SKILL** needed for SMA: KNOWING how to ITERATIVELY LEARN
• Success determined by ability to fail & adaptively learn
• Hyper-learning & innovation economy: pace of learning & innovation going nonlinear w/ asymptotic complexity


• half-life of engineering degree in late 1920’s ~35 years;
• a degree from 1960 ~10 years

2002, William Wulf, president of the National Academy of Engineering

• “half-life of engineering knowledge... is 7 to 2½ years.”
• more recent estimates: low end of range, especially for those working in IT.

2008: Kruchten conjectured in a paper for IEEE Software that

• half-life of software engineering ideas: ~5 years.

https://spectrum.ieee.org/riskfactor/computing/it/an-engineering-career-only-a-young-persons-game

What is knowledge shelf-life today?

https://learningspy.co.uk/learning/o-brave-new-world-search-21st-century-qualifications

https://blog.ekaplus.com/blog/advanced-analytics-can-help-you-manage-the-knowledge-doubling-curve/
Perceived Importance of Skills Today

Higher Cognitive
- Critical Thinking
- Creativity
- PM

Social/EQ
- Leadership
- Communication
- Interpositional
- Empathy
- Adaptability
- Teaching
- Entrepreneurship

Higher Cognitive
- Advanced IT
- Basic Digital
- Tech Design
- Data Analytics
- Data Analytics
- Teaching
- Entrepreneurship

Skills:
- Physical & Manual
- Basic Cognitive

Expected Future Skill Need

Skills needed less of in the future
- Basic Data Input
- Equipment Operation
- Basic Literacy
- Craft & Technician
- Gross Motor
- Complex Information Processing
- Scientific Research & Development

Skills needed more of in the future
- Leadership
- Communications & Negotiation
- Interpositional
- Inverntorship
- Entrepreneurship
- Teaching & Training
- Advanced Data Analyst
- Advanced Literacy
- Adaptability
What got us here, won’t get us there.

- Lack of recognition. Failure to recognize inflection point; failure to respond
- Panic paralysis. Fear of missteps, inability to keep clear head deters productive call to action.
- Old Habits. Falling into the same patterns that have gotten you through in the past feels safe, but it’s in no way an actual progression for your company.
Digital Talent Cutting-Edge Knowledge Workers Want an Environment Affording:

- **Mastery | Autonomy | Purpose** | entrepreneurialism (creating under extreme uncertainty)
- **Culture of innovation** = collaboration = creating something new
- **Learning**: the scientific method – everything is a hypotheses. A/B testing, hyper improvement processes
- **Design Thinking** – empathy, pain points / insights, ideate, convergent / divergent thinking, experiment, repeat
Digital Talent Cutting-Edge Knowledge Workers Want an Environment Affording:

- **Mastery | Autonomy | Purpose | Entrepreneurialism** (creating under extreme uncertainty)
- **Culture of innovation** = collaboration = creating something new
- **Learning**: the scientific method – everything is a hypotheses. A/B testing, hyper improvement processes
- **Design Thinking** – empathy, pain points / insights, ideate, convergent / divergent thinking, experiment, repeat
AGILE ACQUISITION

Why are we here?

Deliver performance at the speed of relevance

Streamline rapid, iterative approaches from development to fielding

When does learning occur?

2011
“No plan survives first contact with the enemy. What matters is how quickly the leader is able to adapt.” – Tim Hartford, Adapt: Why Success Always Starts with Failure

1996
“Everybody has a plan until they get punched in the mouth.” – Mike Tyson

Mid 20th century
“No plan of operations reaches with any certainty beyond the first encounter with the enemy’s main force.” – Dwight Eisenhower

500 B.C.
“Those who are victorious plan effectively & change decisively.” – Sun Tzu
Agile Manifesto

Through this work we have come to value:

- Individuals and interactions
- Working software
- Customer collaboration
- Responding to change

- Processes and tools
- Comprehensive documentation
- Contract negotiation
- Following a plan

That is, while there is value in the items on the right, we value the items on the left more.

The manifesto is often misinterpreted to mean:

- no documentation,
- no process, and
- no plan!

http://www.agilemanifesto.org/
AGILE BS
CI/CD: AGILE VS? DEVSECOPS

Waterfall
- Design
- Code
- Test
- Deploy

Agile
- Design
- Code & Test
- Deploy
- Code & Test

DevSecOps
- Design
- Continuous Operations
- Continuous Deployment
- Continuous Delivery
- Continuous Integration
- Lean
- Agile
- Waterfall
DevOps shortens the release period

Removes human intervention and handoffs

Improves reliability & security
WHERE IS THE SEC IN DEVSECOPS?

Secret Detection
Leak Prevention

Artifact Scanning
STIGs

Persistent Backlog and Design
Imperatives

Dynamic Application Analysis
Hardened Containers

Operational Metrics
Logging & Auditing

Static Code Analysis
Secure Coding

Fuzz Testing
Penetration Testing

DevSecOps mindset: "everyone is responsible for security"
Start with clear goals and be positioned to deliver value

Technical practices: continuous integration, continuous delivery, and automated testing

Cultural practices: rapidly receive and take action on user feedback; a low-blame environment (in post-mortems)

Use of available industry-standard tools to accomplish the above

100 containerized products
Centrally accredited in containers
Bulk licenses
Continuous monitoring of the cyberstack (Devsecops
DevSecOps engineers
(Contract / Repo in work)
Open source container management
Kill container – every 4 hours
Rolling updates – no downtime

Container: build & authorize once, run anywhere (continuous monitoring in production)
OLD WAY: SOFTWARE ENGINEERING VS. OPERATIONS

Conway’s Law: “How to organize our teams affects how we perform our work”
- Share common goals from top to bottom
- Enable business value-oriented team
- Functional team
- Share responsibilities (e.g., security is everyone’s job)
- Keep team size small (Amazon 2 pizza rule)

Why?
Support shift left?

Pre-configured for pernicious “Us versus Them” and “not my job” culture that will emerge
Everyone has a process. Map the process. Make security part of that process mapping. AO understand how automation achieves his ATO requirements.

Typical product workflow where software engineering throws the product over the wall to operations pernicious “Us versus Them” culture can emerge.
NEW WAY: SOFTWARE ENGINEERING & OPERATIONS (DEVSECOOPS)

DoD needs (1) Security and (2) Reliability to enable (3) Increased Velocity and rapidly deliver capability to the Warfighter

Site Reliability Engineering + Conway’s Law = Build together. Deliver together. Unleash innovation and enable the fight together.
Continuous Integration Metrics

- **Code and Automated Build/Release**
  - Build Automation (% steps automated)
  - Average Builds per Day/week (by pass, fail)
  - Duration per Build (minimum, maximum, average)

- **Development Test**
  - Unit Test Coverage (% automated, % coverage)
  - Static Code Analysis Coverage (% automated, % coverage)
  - Functional Thread Test Coverage (% automated, % coverage)

- **System Integration**
  - Integrated Build Frequency by Pass, Fail (# of deployments per day/week)
  - Integrated Build Recovery (average time between failed deployment and system restored to good state)
  - For Cloud, Enterprise and other Compute-intensive Systems
    - Change Volume, in time-series (deployed story points, ESLOC, etc.)
    - Lead Time, in time-series (time from development to deployment)

- **Cyber Monitoring (security controls & patches)**
  - % of automated logging, monitoring and auditing of cybersecurity controls

Agile and Continuous Integration (CI) are critical to minimize program risk
### DEVOPS METRICS

<table>
<thead>
<tr>
<th>Survey questions</th>
<th>High IT performers</th>
<th>Medium IT performers</th>
<th>Low IT performers</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Deployment frequency</strong>&lt;br&gt;For the primary application or service you work on, how often does your organization deploy code?</td>
<td>On demand (multiple deploys per day)</td>
<td>Between once per week and once per month</td>
<td>Between once per week and once per month*</td>
</tr>
<tr>
<td><strong>Lead time for changes</strong>&lt;br&gt;For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code commit to code successfully running in production)?</td>
<td>Less than one hour</td>
<td>Between one week and one month</td>
<td>Between one week and one month*</td>
</tr>
<tr>
<td><strong>Mean time to recover (MTTR)</strong>&lt;br&gt;For the primary application or service you work on, how long does it generally take to restore service when a service incident occurs (e.g., unplanned outage, service impairment)?</td>
<td>Less than one hour</td>
<td>Less than one day</td>
<td>Between one day and one week</td>
</tr>
<tr>
<td><strong>Change failure rate</strong>&lt;br&gt;For the primary application or service you work on, what percentage of changes results either in degraded service or subsequently requires remediation (e.g., leads to service impairment, service outage, requires a hotfix, rollback, fix forward, patch)?</td>
<td>0-15%</td>
<td>0-15%</td>
<td>31-45%</td>
</tr>
</tbody>
</table>

* Note: Low performers were lower on average (at a statistically significant level), but had the same median as the medium performers.

+ Cyber Debt
+ Technical Debt

From *Accelerate* by Nicole Forsgren, PhD, Jez Humble, and Gene Kim
How long does it take to build a new culture?

What’s the fastest way to deter a new culture?

Not a sheer technical challenge!

Building an ecosystem!

Lack of leadership and strategy communication will kill transformation.

Scaling bad Agile will lead to failure.

Start small; attack key pain points; and scale.

DO LOOP: SOFTWARE IS EATING THE WORLD. CULTURE EATS OUR SW STRATEGY FOR BREAKFAST.

HOW FAST CAN I BOLT ON DEVSECOPS?

I GOTTA HAVE MORE DEVSECOPS.
“VUCA environments impede a leader’s ability to understand, to decide, to communicate and ultimately to act decisively – a prerequisite for effective action in war (and business).

Only a few leaders were able to fight through all the complexity and uncertainty and chart a way forward for their organizations. They imposed their wills on these most complex environments and succeeded where others didn’t. These were the leaders that made a difference for the mission.” – General George W. Casey, Jr.

“Transforming large organizations is hard. In our experience, it has typically taken five to 10 years to scale a transformation. However, technology is advancing at a much faster pace than that. In today’s world, digital transformations need to be substantially accomplished much faster.” - Edward Hess

“The record of studies on digital transformation indicate a high failure rate, with a notable 2013 McKinsey study finding that 70% fail. ...the biggest problem is the mind-set. ...where most companies go wrong.”

...digital transformation: not an event that happens. It’s a journey with a road that never ends. will continue – potentially indefinitely, but certainly for three to five years or longer.

2016: expanding counter-threat mission necessitated an evolving digital transformation to respond to the warfighter’s “tactical edge” Latest Time of Value (LTOV) needs in hours versus days or weeks.

Solution

- **Reorganized teams:** JIDO integrated its people across engineering & cybersecurity teams.
- **New Culture:** continuous improvement, achieving common focus, and shared priorities.
- **Setup the Pipeline & Shift Left:**
  - On the technology side, JIDO developed a continuous integration / continuous deployment (CI/CD) pipeline and “shifted security to the left”
  - JIDO developers and operations engineers deliver fully tested, commit-level increments of new code on a security-hardened, patched, and approved infrastructure enclave.
- **Automation:** when automating nearly all the security controls associated with the STIG, JIDO has defined a risk-managed software delivery pipeline
  - Vetting the pipeline & the process to give confidence, consistency w/ recent test results;
  - Minimizes manual (months long) human review of every software update or modification;
  - This process provides a real-time view of the systems, networks, and vulnerabilities to the Authorizing Official (AO) while delivering immediate value to operational warfighters.
CASE STUDY – JIDO

Win Win: Impact goes beyond raw benefit of the shift left (security and testing)

- **Energized, Multidisciplinary teams:** Team synergy increased due composition of varied but complimentary experience, increased interaction between teams, daily forced communication, qualifications, and skills.
  - **Job satisfaction** among participants increased due to visibility of impact.
  - **Shared Focus:** Teams began working in a focused manner sharing communications, common priorities and working towards common goals.

- **Trust:** Security not viewed as adversary but as valued cooperative:
  - An organization will never know the disasters that did not occur.
  - Now, the team has a real time understanding of vulnerabilities in its custom code and a rapid way to respond.
  - Therefore, the team can patch custom code the same day a vulnerability is observed where previously it could take weeks or even months.
  - The AO also has an improved understanding of technical risks on the network with transparent dashboards verse static reports.

Team of Teams: Empowered Execution, Common Purpose, Shared Consciousness, Trust
# DevOps Metrics

## Survey Questions

<table>
<thead>
<tr>
<th>Survey questions</th>
<th>High IT performers</th>
<th>Medium IT performers</th>
<th>Low IT performers</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Deployment Frequency</strong>&lt;br&gt;For the primary application or service you work on, how often does your organization deploy code?</td>
<td>On demand (multiple deploys per day)</td>
<td>Between once per week and once per month</td>
<td>Between once per week and once per month*</td>
</tr>
<tr>
<td><strong>Lead Time for Changes</strong>&lt;br&gt;For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code commit to code successfully running in production)?</td>
<td>Less than one hour</td>
<td>Between one week and one month</td>
<td>Between one week and one month*</td>
</tr>
<tr>
<td><strong>Mean Time to Recover (MTTR)</strong>&lt;br&gt;For the primary application or service you work on, how long does it generally take to restore service when a service incident occurs (e.g., unplanned outage, service impairment)?</td>
<td>Less than one hour</td>
<td>Less than one day</td>
<td>Between one day and one week</td>
</tr>
<tr>
<td><strong>Change Failure Rate</strong>&lt;br&gt;For the primary application or service you work on, what percentage of changes results either in degraded service or subsequently requires remediation (e.g., leads to service impairment, service outage, requires a hotfix, rollback, fix forward, patch)?</td>
<td>0-15%</td>
<td>0-15%</td>
<td>31-45%</td>
</tr>
</tbody>
</table>

* Note: Low performers were lower on average (at a statistically significant level), but had the same median as the medium performers.
CASE STUDY - JOINT IMPROVISED-THREAT DEFEAT ORGANIZATION (JIDO)

Quantitative Impact

Over a 6-month period, JIDO measured Key Performance Indicator (KPI) impact against pre- and post-DevSecOps enablement:

<table>
<thead>
<tr>
<th>KPI</th>
<th>Definition</th>
<th>Legacy</th>
<th>DevSecOps Enabled</th>
<th>%/$ Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Acceptable Quality Level (AQL)</td>
<td>Service Level Acceptable Quality Level (AQL) for Average Operational Availability of services</td>
<td>99.5%</td>
<td>99.9%</td>
<td>+3 HRS MONTHLY UPTIME</td>
</tr>
<tr>
<td>Continuous Authorization</td>
<td>Average time to complete code deployment after initial A&amp;A</td>
<td>23 Days</td>
<td>6 Hours</td>
<td>92% FASTER</td>
</tr>
<tr>
<td>Deployment Frequency</td>
<td>The frequency new code reach customers</td>
<td>11</td>
<td>98 Releases</td>
<td>891% INCREASE</td>
</tr>
<tr>
<td>Initial System Authorization</td>
<td>Cybersecurity risk assessment threshold determination for pipeline including major system design and compliance with DoD Risk Management Framework</td>
<td>12 Months</td>
<td>3 Month</td>
<td>75% REDUCTION</td>
</tr>
<tr>
<td>Lead Time Reduction</td>
<td>The time from the start of a development cycle (the first new code) to deployment is the change lead time</td>
<td>169.83 Days</td>
<td>12 Days</td>
<td>93% REDUCTION</td>
</tr>
<tr>
<td>Mean Time to Provision</td>
<td>The average time it takes to add additional services to an environment</td>
<td>6 Months</td>
<td>2 Hours</td>
<td>99.79% REDUCTION</td>
</tr>
<tr>
<td>Mean Time to Recovery</td>
<td>The average time from deployment failure to recovery</td>
<td>15.5 Minutes</td>
<td>4 Minutes</td>
<td>74% REDUCTION</td>
</tr>
<tr>
<td>Operating Cost</td>
<td>Change in operating costs based on leveraging open source tooling vs legacy COTS dependent architecture</td>
<td>$1.8M</td>
<td>$150K</td>
<td>91.66% REDUCTION</td>
</tr>
</tbody>
</table>

KPI | Result
---|---
Deployment Frequency Increase | 891%
Availability AQL | 3HR (MONTHLY)
Lead Time | 93%
MTTR | 74%
Initial System Authorization | 75%
The number of vulnerabilities and number of attacks have grown at a near exponential rate.

The time between new vulnerabilities being discovered is dropping to below a week.

The ability to update and test code in days versus weeks is essential to maintaining cybersecurity.

Can we afford to continue the old way of doing business in an age of ever increasing Zero Day vulnerabilities?
Need for New Business Model

Near peers quickly iterating, F-22 requires rapid changes to maintain edge.
F-22 Capability Pipeline

Cannot scope Continuous Integration / Continuous Delivery (CI/CD) model by capabilities which are thinly sliced and overlapping.
DEVSECOPS ACADEMY

- THE WHY
- FAMILIARITY
- JUST ENOUGH HANDS-ON
- GROWTH & LEARNING MINDSET
Backup
LOE C -- New Paths for Digital Talent

LOE A: new acquisition pathways
LOE B: utilizing digital infrastructure

LOE C: software development as a high-visibility, high-priority career track and increasing the level of understanding of modern software within the acquisition workforce.

**Recommendation C1.** Create software development units in each Service consisting of military and civilian personnel who develop and deploy software to the field using DevSecOps practices.

**Recommendation C2.** Expand use of (specialized) training programs for CIOs, SAEs, PEOs, and PMs that provide (hands-on) insight into modern software development (e.g., agile, DevOps, DevSecOps) and the authorities available to enable rapid acquisition of software.

**Recommendation C3.** Increase the knowledge, expertise, and flexibility in program offices related to modern software development practices to improve the ability of program offices to take advantage of software centric approaches to acquisition.

LOE D: new practices (acquire entire SSE; DevSecOps)
Accelerate adoption of modern SW delivery in DoD:

- Build a powerful network to maximize DoD/DAU's "enterprise effect."

- Establish the Defense Software Alliance (DSA) at DAU -- cuts across DoD silos.
  - Engagement platform supporting (a) DoD Innovation Lab (b) DoD DevSecOps Academy graduates

- The DSA is a modern "community of practice" focused on
  a. its members sharing a *required set of knowledge*
  b. crucially, facilitates engagement / active sharing (e.g. "Secure Slack/Rocket.Chat for DoD")
  c. gathers pain points / insights / and develops solutions across DoD

- DoD employee gains membership to the DSA network
  - upon understanding knowledge related to modern SW practice -- within DoD context.
  - gain full access if they complete modern SW course(s) in multiple DAU tracks
  - **Tracks/paths:** "DoD DevSecOps Leader" or "DoD Agile Leader" // "Agile Contracts Leader" "DoD Cyber Leader" // "DoD SRE Leader" // "DoD Cloud Architect" // "Weapons SW Leader"

- Growth of connections in this network goes quadratic.
- More connections -- more collective problem solving and knowledge sharing.
- Higher the number of "initiated" connections the better the network, outcomes, and feedback to DAU

- The more valuable the network, the more demand for courses/workshops -- and "certifications" -- in-turn improving the courses -- **virtuous cycle repeats**

- **DSA connects the entire DoD** -- and to DAU, as the focal point for lessons learned / pain points / best practices

- DoD’s leading DevSecOps innovators will **set the initial value of the DSA kernel**
<table>
<thead>
<tr>
<th>ON GROUNDS DAY 1 WORKSHOP</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The Why</td>
</tr>
<tr>
<td>o State of Play in DoD</td>
</tr>
<tr>
<td>o Why do we Need to Change?</td>
</tr>
<tr>
<td>o What is DevSecOps?</td>
</tr>
<tr>
<td>o Value Proposition</td>
</tr>
<tr>
<td>o Challenges and Constraints in DoD (Warfighting Domain overviews)</td>
</tr>
<tr>
<td>• Leadership and Management</td>
</tr>
<tr>
<td>o How and what do we change?</td>
</tr>
<tr>
<td>o Role of Leadership</td>
</tr>
<tr>
<td>o Transformation Strategy and Roadmap</td>
</tr>
<tr>
<td>o Ecosystem and Governance</td>
</tr>
<tr>
<td>o People and Culture (Way of Life, Organizing &amp; Staffing in PMO &amp; KTR)</td>
</tr>
<tr>
<td>o Engagement Strategy</td>
</tr>
<tr>
<td>o Acquisition &amp; Sustainment Strategy (Services? SW Factory?)</td>
</tr>
<tr>
<td>o Contracting for DevSecOps / RFP / Incentives / Data Rights</td>
</tr>
<tr>
<td>• Tech and Implementation</td>
</tr>
<tr>
<td>o Cybersecurity: RMF and cATO process</td>
</tr>
<tr>
<td>o Solution Architecture and Design</td>
</tr>
<tr>
<td>o Enablers (resilience, availability, modularity)</td>
</tr>
<tr>
<td>o Cloud nexus with DevSecOps</td>
</tr>
<tr>
<td>o Domain &amp; Platform Considerations (DDIL; Safety-Critical; DO-178C)</td>
</tr>
<tr>
<td>o Legacy System Modernization/Strangler Pattern</td>
</tr>
<tr>
<td>• Software Development Process</td>
</tr>
<tr>
<td>o Lean Startup / Design Thinking (pain points and value stream mapping)</td>
</tr>
<tr>
<td>o Requirements (JICDS)</td>
</tr>
<tr>
<td>o Shift Left: DT/OT</td>
</tr>
<tr>
<td>o Systems Engineering Technical Reviews (Navigation &amp; Expectations)</td>
</tr>
<tr>
<td>o Metrics &amp; DevSecOps Maturity; DevSecOps BS</td>
</tr>
<tr>
<td>• Special Topics</td>
</tr>
<tr>
<td>o DoD SevSecOps Enterprise Platform (DSOP)</td>
</tr>
<tr>
<td>o Case Study</td>
</tr>
<tr>
<td>o Day in the Life of a DoD SW Factory</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ON GROUNDS DAY 2 - HANDS-ON</th>
</tr>
</thead>
<tbody>
<tr>
<td>• What is DevOps?</td>
</tr>
<tr>
<td>• Organizational Needs and linking Business into DevOps</td>
</tr>
<tr>
<td>• Secure DevOps</td>
</tr>
<tr>
<td>o DevOps Pipeline Security,</td>
</tr>
<tr>
<td>o Application Security</td>
</tr>
<tr>
<td>o Security activities and automation</td>
</tr>
<tr>
<td>• Communication and Collaboration</td>
</tr>
<tr>
<td>o Security culture</td>
</tr>
<tr>
<td>o Effective communication amongst all stakeholders.</td>
</tr>
<tr>
<td>o Micro learning culture on security</td>
</tr>
<tr>
<td>• Infrastructure as Code</td>
</tr>
<tr>
<td>o Environment hardening</td>
</tr>
<tr>
<td>o Compliance check with IaC</td>
</tr>
<tr>
<td>o First step to RMF/ATO</td>
</tr>
<tr>
<td>• Continuous Integration &amp; Testing</td>
</tr>
<tr>
<td>o Automated Security Testing,</td>
</tr>
<tr>
<td>o Application specific penetrating testing</td>
</tr>
<tr>
<td>o Various Gateways on security testing</td>
</tr>
<tr>
<td>• Continuous Delivery/Deployment</td>
</tr>
<tr>
<td>o Container Security</td>
</tr>
<tr>
<td>o Microservices, Containers &amp; Orchestration (K8)</td>
</tr>
<tr>
<td>o Authenticity of build and dependencies</td>
</tr>
<tr>
<td>o Secure Deployment pipeline</td>
</tr>
<tr>
<td>• Process Monitoring and Measurement</td>
</tr>
<tr>
<td>o What are the security metrics</td>
</tr>
<tr>
<td>o Where to collect and how to monitor them</td>
</tr>
<tr>
<td>o Workshop Summary and Q/A</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ASYNC LEARNING – VIDEO SERIES</th>
</tr>
</thead>
<tbody>
<tr>
<td>DoD DevSecOps Platform: TECH DEEP DIVE</td>
</tr>
<tr>
<td>Featuring: Nicolas Chaillan</td>
</tr>
<tr>
<td>o CI/CD introduction and stack which includes Kubernetes/Containers best practices</td>
</tr>
<tr>
<td>o Container hardening process</td>
</tr>
<tr>
<td>o Cybersecurity stack and side car container (3H)</td>
</tr>
<tr>
<td>o Microservices architecture and best practices</td>
</tr>
<tr>
<td>o Strangler pattern</td>
</tr>
</tbody>
</table>

Leading Transformation and Culture - People and Tech Roadmaps
Featuring: Leo Garciga
- The Why
- Ecosystem and Governance
- Acquisition Strategy
- RMF and CATO
INNOVATION & LEARNING

- Automation, DevSecOps, AI and ML: operational efficiencies will be commoditized / table stakes
- Differentiator: Quality of thinking; Ability to ideate, relate & collaborate
- DoD starting place – we have a lot to learn
- Learning and Transformation Inhibitors
  - Fear – room to change and fail
  - Fixed mindset – we’ve always done it this way
“Simply delivering what was initially required on cost and schedule can lead to failure in achieving our evolving national security mission — the reason defense acquisition exists in the first place.”

Honorable Frank Kendall Under Secretary of Defense (AT&L) 2015 Performance of The Defense Acquisition System
DEVOPS

- **DevOps** is a *software engineering culture* and practice that aims at *collaboration & unifying* software development (Dev) and software operation (Ops).
- Main characteristic of the DevOps movement: *strongly advocate automation and monitoring* at all steps of software construction, from integration, testing, releasing to deployment and infrastructure management.
- DevOps aims at shorter development cycles, *increased deployment frequency, earlier defect discovery*, and more secure and reliable releases, in close alignment with business objectives.

DevOps is **NOT ENOUGH**!
**DevSecOps** is needed to Shift Left w/ the cybersecurity stack built-in to the DevOps pipeline.

Source: OUSD R&E Systems Engineering, Aggregation of typical DevOps tool stack components found in and around the DoD
WHERE IS THE SEC IN DEVSECOPS?

DevSecOps Proposed Architecture*

DevOps Platform + Sidecar = DevSecOps
(Sidecar is key to the cyberstack; bakes Sec in):
- Visibility into SW assets and security state
- Push logs to monitoring tools / telemetry
- Log pooling enables behavior detections across the portfolio
- Persistent scans of containers; not manual every 6 months
- Access control (zero trust whitelisting; cannot laterally hack into other systems)
• **Contracting for DevSecOps** – Include language within RFP/SOW Sections
  L&M and Acquisition Strategy
  • Include DevSecOps within ICD/CDD, TPMs/KSAs
• **Staff Organization & Skills** – Evaluate staff organization (e.g., teams) and technical skill levels to ensure sufficiency. Augment as necessary
• **Ecosystem & Governance** – People, processes & technology enabled through collaboration, automation and analysis
• **DAU Training and Workshops** – train the teams that will be implementing.
DevSecOps Academy (Workshop and Consulting)

DoD Software Alliance

Network Effect and Virtuous Cycle: Scaled and Supported DevSecOps Factories Across DoD Feedback Loops / Leverage Entire DoD SW Talent Pool
DOD INNOVATION LAB // VIRTUAL TRAINING RANGE >> LEARNING FLOW

- (a) [804 boot camp] > (b) [Design Thinking Workshop] > (c) [DevSecOps – Virtual Training Lab]

- ...connecting the entire Digital Product Delivery pipeline:

- (a) incubating innovative acquisition strategies & creative compliance (OTA/804) -- e.g., acquiring the SW factory

- (b) prioritizing MVP requirements (Design Thinking) in an entrepreneurial way in the factory

- (c) delivering those prioritized requirements (deploy code) in a modern DevOps pipeline / virtual training range
Elevator Pitch

DoD lacks modern software development competency and practices. Manual build/release/test has cost the taxpayer billions and is delaying time-to-Warfighter. DoD already challenged to deliver SW. Demand for SW will only grow. DoD needs secure, reliable, rapid software delivery.

Algorithmic warfare is the future. Winning the fight anywhere -- demands software everywhere.

The DoD has embraced Agile (a nearly 20 year old “fad”) as a means to shorten the development cycle and ensure programs deliver the right capabilities to the user.

DevSecOps enhances this to increase reliability of the system, allow changes to be rapidly developed and deployed, and allow security testing and patches/updates in a continuous integration environment with the ability to deploy on demand.

DevSecOps workshop will provide the ability for programs to start there transformation to DevSecOps – and allow the DoD to deliver capabilities securely, reliably and rapidly to Warfighter in potentially hours instead of months or years.
Mr. Sean Brady serves as the Learning Director for Software Acquisition at the Defense Acquisition University. He leads strategy to transform DoD’s practices, competencies, training, and workforce—and accelerate the adoption of modern, commercial software development practices across DoD and the DAU curriculum.

Prior to DAU, Mr. Brady served 9 years as the Deputy Director for Software Engineering (SWE), in the Office of the Deputy Assistant Secretary of Defense for Systems Engineering (DASD(SE)), within the Office of the Secretary of Defense (OSD). He led change, provided oversight of 170+ major programs (a $1.7T portfolio), and managed governance of Software Acquisition and Development within DoD. He led SWE oversight on DoD's most complex, highest-visibility defense programs (RD&E > $480M; procurement > $2.79 billion). Mr. Brady is an expert in software parametric statistical analysis and assessing large-scale Agile software development efforts. He informed DoD’s senior-decision makers, industry CEOs, and Congress on SWE across Army, Navy, Marine Corps and Air Force programs. His strategic duties include leading policy and guidance development; workforce planning; and outreach to optimize the DoD’s SWE capability. He launched DoD’s largest acquisition workforce modernization initiative (impacting 200K+ professionals). In addition, he championed OSD’s efforts to improve performance measurement practices across DoD and industry.

Prior to his role in the Pentagon, he served in the Army's RDECOM/ARDEC as an Armament Software Engineering Center (ASEC) Special Projects Team Lead and as a Program Manager, Close Combat Systems Project Officer (PO) where he planned and executed high-visibility experimental and rapid fielding programs -- supporting elite special operations and front-line Warfighters. He is the Defense Innovation Board’s Software Acquisition workforce co-lead, a member of the Army’s Acquisition Corps and has served as a US Delegate to NATO. Mr. Brady holds a Bachelor of Science in Computer and Electrical Engineering from Rutgers University; a Master of Science in Quantitative Software Engineering from Stevens Institute of Technology; is an Executive MBA candidate from the University of Virginia (Class of 2020); and holds a graduate certificate in Entrepreneurship and Innovation from Stanford University.