Today’s Presenters – *DevSecOps Assessment and Maturity Model*

**John Meyer**  
*Solutions Architect, ASMGi*  
jmeyer@asmgi.com

**Frank Yako**  
*CIO, Director of Strategic Initiatives*  
fyako@asmgi.com
Agenda

- What does it mean to “Move to the Left”?
- Introduction and Objectives
- Overview of SDLC
- What is DevOps why DevOps was Adopted?
- DevOps to DevSecOps
- Methods to determine DevSecOps maturity
- How Does an Organization Assess DevSecOps Maturity and Apply It?
What does it mean to “Move to the Left”? 

Security moves from an afterthought to the forefront (e.g. “To the Left”) thus

DevSecOps
Introduction and Objectives

◆ Provide an Overview and Introduction to DevOps and DevSecOps
◆ Discuss the Reasons why an Organization would Consider adopting DevSecOps
◆ Compare the Differences between DevOps and DevSecOps
◆ Explore How to begin the “Move to the Left”
◆ How an Organization integrates DevSecOps and Matures
The Legacy Software Development Lifecycle

Software Development Methodologies

**Use cases:**
- Simple small or mid-sized projects with clearly defined and unchanging requirements (small company website development).
- Projects with the need for stricter control, predictable budget and timelines (e.g., governmental projects).
- Projects that must adhere to multiple rules and regulations (healthcare projects).
- Projects where a well-known technology stack and tools are used.
- Projects where failures and downtimes are unacceptable (e.g., medical software, aviation fleet management software).

**Use cases:**
- Projects with the need for stricter control, predictable budget and timelines (e.g., governmental projects).
- Projects that must adhere to multiple rules and regulations (healthcare projects).
- Projects where a well-known technology stack and tools are used.
- Projects where failures and downtimes are unacceptable (e.g., medical software, aviation fleet management software).

Source: ScienceSoft 2020
Alternative Software Development Methodologies

Use cases:

Large, mission-critical enterprise applications that preferably consist of loosely coupled parts, such as microservices or web services.
The Modern Software Development Lifecycle

Software Development Methodologies

**SCRUM**

- **Use cases:** Practically any startup initiatives, when end users’ early feedback is required.
- **Use cases:** Projects with unclear business needs or too ambitious/innovative requirements.
- **Use cases:** Projects that are large and complicated.
- **Use cases:** Large projects that are easy to divide into small functional parts and can be developed incrementally over each iteration.

**SPRINT 1**
2-4 WEEKS

**SPRINT 2**
2-4 WEEKS

**SPRINT 3**

**SPRINT 4**

**Risk analysis and alternatives evaluation**

1. **Determination of objectives, alternatives, and constraints**
2. **Development of deliverables**
3. **Next iteration planning**

Source: ScienceSoft 2020
Developers and Operations were first to the Scene...

DevOps

Developer

Operations
Why was DevOps adopted?

<table>
<thead>
<tr>
<th>Objective</th>
<th>1st Choice</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
<th>5th</th>
<th>Sum</th>
<th>Very Satisfied (6-7) n = 139</th>
<th>Less Satisfied (1-5) n = 105</th>
<th>Very Successful at Scaling DevOps (6-7) n = 136</th>
<th>Less Successful at Scaling DevOps (1-5) n = 107</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improving Agility</td>
<td>15%</td>
<td>12%</td>
<td>14%</td>
<td>10%</td>
<td>8%</td>
<td>59%</td>
<td>63%</td>
<td>54%</td>
<td>62%</td>
<td>55%</td>
</tr>
<tr>
<td>Improving Release/ Fewer Defects</td>
<td>10%</td>
<td>10%</td>
<td>12%</td>
<td>11%</td>
<td>9%</td>
<td>53%</td>
<td>49%</td>
<td>60%</td>
<td>51%</td>
<td>57%</td>
</tr>
<tr>
<td>Improving System Reliability</td>
<td>12%</td>
<td>11%</td>
<td>10%</td>
<td>8%</td>
<td>11%</td>
<td>52%</td>
<td>51%</td>
<td>56%</td>
<td>50%</td>
<td>56%</td>
</tr>
<tr>
<td>Reduction in Overall Waste</td>
<td>11%</td>
<td>9%</td>
<td>10%</td>
<td>10%</td>
<td>10%</td>
<td>51%</td>
<td>54%</td>
<td>46%</td>
<td>58%</td>
<td>40%</td>
</tr>
<tr>
<td>Cost Reduction</td>
<td>10%</td>
<td>12%</td>
<td>12%</td>
<td>9%</td>
<td>7%</td>
<td>49%</td>
<td>42%</td>
<td>56%</td>
<td>40%</td>
<td>57%</td>
</tr>
<tr>
<td>Fast Delivery of Customer Value</td>
<td>9%</td>
<td>14%</td>
<td>11%</td>
<td>7%</td>
<td>7%</td>
<td>48%</td>
<td>47%</td>
<td>47%</td>
<td>43%</td>
<td>45%</td>
</tr>
<tr>
<td>Improving User Experience</td>
<td>8%</td>
<td>7%</td>
<td>10%</td>
<td>11%</td>
<td>12%</td>
<td>47%</td>
<td>50%</td>
<td>46%</td>
<td>51%</td>
<td>47%</td>
</tr>
<tr>
<td>Maintaining Market Competitiveness</td>
<td>5%</td>
<td>10%</td>
<td>7%</td>
<td>8%</td>
<td>10%</td>
<td>41%</td>
<td>45%</td>
<td>37%</td>
<td>42%</td>
<td>42%</td>
</tr>
<tr>
<td>More Predictable Delivery of Customer Value</td>
<td>9%</td>
<td>8%</td>
<td>7%</td>
<td>7%</td>
<td>7%</td>
<td>38%</td>
<td>40%</td>
<td>37%</td>
<td>38%</td>
<td>40%</td>
</tr>
<tr>
<td>Improving IT Employee Job Satisfaction</td>
<td>10%</td>
<td>6%</td>
<td>4%</td>
<td>7%</td>
<td>6%</td>
<td>33%</td>
<td>40%</td>
<td>27%</td>
<td>39%</td>
<td>32%</td>
</tr>
<tr>
<td>No Definitive Goals Were Set</td>
<td>0.37%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1%</td>
<td>-</td>
<td>1%</td>
<td>1%</td>
</tr>
</tbody>
</table>

n = 273

Q: What were the top objectives for your organization’s DevOps efforts at the beginning? Summary of Top 5 Ranks (1-5)
Source: Gartner 2019 DevOps Survey
ID: 377263

*Statistical significant difference: 95%*
...and then Security appeared on the Scene

DevSecOps

Developer

Security

Operations
DevSecOps Software Development Lifecycle

The DevSecOps Toolchain

1. Address Technical Debt, DevSec Metrics, Threat Modeling, Security Tool Training
2. IDE Security Plug-Ins
3. SAST/DAST/IAST, SCA
4. Chaos Monkey, Input Fuzzing, Integration Test
5. Software Signing
6. Signature Verify, Integrity Checks, Defense In-Depth Measures
7. RASP, UEB/A Network Monitoring, Penetration Test
8. Security Orchestration, RASP/WAF Shielding, Obfuscation
9. Dev Consumable, Correlated Vulnerability Analysis, IoC/TI STIX TAXII
10. Security Technical Debt, Modify Incident Response, Modify DND

Source: Gartner
ID: 377293
Regulations and Contractual Obligations Under Which DevOps Practitioners Are Working

<table>
<thead>
<tr>
<th>Regulations and Contractual Obligations</th>
<th>Percentage of Respondents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Privacy Laws and Regulations, Differs by Region (for e.g., EU GDPR)</td>
<td>47%</td>
</tr>
<tr>
<td>ISO 27001: Information Security Management System</td>
<td>46%</td>
</tr>
<tr>
<td>Federal Information Security Management Act (FISMA)</td>
<td>41%</td>
</tr>
<tr>
<td>Health Insurance Portability and Accountability Act (HIPAA)</td>
<td>37%</td>
</tr>
<tr>
<td>Payment Card Industry Data Security Standard (PCI-DSS)</td>
<td>30%</td>
</tr>
<tr>
<td>NIST Cybersecurity Framework (CSF)</td>
<td>29%</td>
</tr>
<tr>
<td>Family Educational Rights and Privacy Act (FERPA)</td>
<td>23%</td>
</tr>
<tr>
<td>Basel or Gramm-Leach-Bliley Act (GLBA)</td>
<td>22%</td>
</tr>
<tr>
<td>Sarbanes-Oxley Act</td>
<td>19%</td>
</tr>
<tr>
<td>Service Organization Control (SOC)2</td>
<td>15%</td>
</tr>
<tr>
<td>Other</td>
<td>1%</td>
</tr>
<tr>
<td>Not Required to Comply With Any Regulations</td>
<td>2%</td>
</tr>
</tbody>
</table>

n = 268 All Respondents, Excluding Unsure

Note: Multiple responses allowed. Mean number calculates excluding "We are not required to comply with any regulations."
What Does it Mean to Mature in DevSecOps

- Finding the right balance and distribution of roles
- Successful integration of security controls and tools throughout the DevSecOps toolchain
- Use DevSecOps practices and automation to substitute traditional controls
- Validate and measure success of DevSecOps compliance efforts by continually monitoring status of compliance
- Scale compliance practices across the organization by leveraging compliance-specific software
How to Begin the Journey

- Understand “Why”. Make the Business justification!
- Assessment = Current State vs. Desired Future State
- Identify Gaps and Propose Solutions
- Build a Roadmap = Can’t Do “IT” all at Once!
- Incremental Change over Time
- Start Now
  - It will take 2 years whether you “Start Now” or start 2 years from now, so Start Now!
What Does an Assessment Look Like

Phase

- Requirements
- Risk Level
- Maturity Level
- Priority

People

- Process
- Technology
- Sub-process
Systematically Assess DevSecOps Maturity

- List phases
- List requirements for each phase
- List processes & sub-processes to satisfy the requirement
- List organization roles needed to satisfy the requirement
- List the technology needs to carry out the processes
- Assess strengths, weaknesses, and risks, then prioritize!
Organization Self-Assessment

DevSecOps Grading & Health Report

- DevSecOps Model Completion: 23/79 (29.11%)
- DevSecOps Risk Assessment: 171/268 (64.21%)
- DevSecOps Maturity: 39/192 (20.08%)

DevSecOps Health Scorecard

- DevSecOps Model Completion: 29.11%
- DevSecOps Risk Assessment: 64.21%
- DevSecOps Maturity: 20.08%
What is Achieved Through a Mature DevSecOps Model

- Improving team dynamics and cross-functionality
- Compliance through automation
- Effectively following business processes
- Meaningfully implementing effective tools
- Iterative reassessment and improvement
Thank You!

800 Superior Ave E, Ste 1050
Cleveland, OH 44114

Phone: 216.255.3040
Fax: 216.274.9647

Email: info@asmgii.com

www.asmgii.com