9th Annual Cyber Resilience Summit

Hosted by:
Dr. Bill Curtis
Mr. Luke McCormack
9TH CISQ CYBER RESILIENCE SUMMIT

OCTOBER 12, 2021

HOSTS:
BILL CURTIS
EXECUTIVE DIRECTOR
CONSORTIUM FOR INFORMATION & SOFTWARE QUALITY

LUKE MCCORMACK
FORMER CIO
DEPARTMENT OF HOMELAND SECURITY
What Is CISQ?

CISQ is chartered to specify measures of software size and quality that can be automated from source code, and promote them through OMG and other international standards organizations.
CISQ’s History of Software Standards

OMG

ISO

Size

Automated Enhancement Points

Automated Function Points

Technical Debt Measure

Reliability Measure

Performance Efficiency Measure

Security Measure

Maintainability Measure

Data Protection Measure

Automated Source Code Quality Measures

ISO/IEC 5055:2021

Automated Source Code Quality Measures

ISO/IEC 19515

Automated Function Points

Automated Source Code Quality Measures
CISQ’s Future Standards

- Software Bill of Materials (SBOM)
- Automated Source Code Data Protection Measure
- Flow and Modernization Measures for Agile/DevOps Environments
- Updated Automated Technical Debt Measure
- Process Maturity Metamodel
- Dependable Programmer Certification
Almost 4000 individual members from Fortune 1000 organizations

Contents:
- Approved standards
- Contract language
- Trustworthy Systems Manifesto
- Presentations
- Webinars
- Tutorials
- Whitepapers
- Use Cases
- Blogs
- News
- Current standards projects
- Process Maturity Metamodel
- Upcoming events

Cyber Resilience Summits
KEYNOTE:
IS TECHNOLOGY THE SOLUTION OR PART OF THE PROBLEM?
TECHNICAL DECISION POINTS ON THE JOURNEY TO RESPONSIBLE COMPUTING

PRESENTED BY:

MARC PETERS
DISTINGUISHED ENGINEER, CTO FOR ENERGY, ENVIRONMENT & UTILITIES EMEA, IBM
GAINING INSIGHT INTO CYBERSECURITY MATURITY

PRESENTED BY:

RON ZAHAVI
CHIEF STRATEGIST FOR IOT STANDARDS, MICROSOFT

MATTHEW JAMES BUTKOVIC
TECHNICAL DIRECTOR, SEI

SAMMY MIGUES
PRINCIPAL SCIENTIST, THE SYNOPSYS SOFTWARE INTEGRITY GROUP
CMMC AND SMM

PRESENTED BY: Ron Zahavi, Chief Strategist for IoT standards, Microsoft Azure IoT, SMM co-author
October 2021
CMMC and SMM

• What are they?
• What’s similar?
• What’s different?
• Complementary use
CMMC and SMM purpose

- **CMMC**: The *Cybersecurity Maturity Model Certification (CMMC)* is a new cybersecurity framework and accompanying certification by the US Department of Defense (DoD). The goal of the new CMMC compliance requirement is to protect Federal Contract Information (FCI) and Controlled Unclassified Information (CUI).

- **SMM**: The Internet of Things (IoT) *Security Maturity Model (SMM)* builds on the concepts identified in the Industrial Internet Security Framework (IISF) and provides a path for IoT providers to understand where they need to be, make intelligent choices about which mechanisms to use and how to invest in the mechanisms to meet their needs.
Structure - CMMC Domains and SMM Practices

- **Governance**
  - Strategy and Governance
  - Threat Modeling and Risk Assessment
  - Supply Chain and Dependencies Management
- **Enablement**
  - Identity and Access Management
  - Asset Protection
  - Data Protection
- **Hardening**
  - Vulnerability and Patch Management
  - Situation Awareness
  - Event and Incident Response, Continuity of Operations

**Domain**
- Security Program Management
- Compliance Management
- Threat Modeling
- Risk Attitude
- Product Supply Chain Risk Management
- Services Third-Party Dependencies Management
- Establishing and Maintaining Identities
- Access Control
- Asset, Change and Configuration Management
- Physical Protection
- Protection Model and Policy for Data
- Implementation of Data Protection Controls
- Vulnerability Assessment
- Patch Management
- Monitoring Practice
- Situation Awareness and Information Sharing
- Event Detection and Response Plan
- Remediation, Recovery and Continuity of Operations

**Subdomain**
- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)
Structure - CMMC Domains and SMM Practices
Structure - CMMC Domains and SMM Practices

Goverance
- Strategy and Governance
- Compliance Management
- Threat Modeling
- Risk Attitude
- Product Supply Chain Risk Management
- Services Third-Party Dependencies Management

Enablement
- Identity and Access Management
- Access Control
- Asset, Change and Configuration Management
- Physical Protection
- Protection Model and Policy for Data
- Implementation of Data Protection Controls

Hardening
- Vulnerability and Patch Management
- Patch Management
- Monitoring Practice
- Situation Awareness and Information Sharing
- Event Detection and Response Plan
- Remediation, Recovery and Continuity of Operations

Domain

Subdomain
- Security Program Management
- Establishing and Maintaining Identities
- Access Control
- Physical Protection
- Vulnerability Assessment
- Patch Management
- Monitoring Practice
- Situation Awareness and Information Sharing
- Event Detection and Response Plan
- Remediation, Recovery and Continuity of Operations

Practice
- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)
Similar topic areas

- Strategy and Governance
- Compliance Management
- Threat Modeling
- Risk Attitude
- Product Supply Chain Risk Management
- Services Third-Party Dependencies Management
- Establishing and Maintaining Identities
- Access Control
- Asset, Change and Configuration Management
- Physical Protection
- Protection Model and Policy for Data
- Implementation of Data Protection Controls
- Vulnerability Assessment
- Patch Management
- Monitoring Practice
- Situation Awareness
- Situation Awareness and Information Sharing
- Event Detection and Response Plan
- Remediation, Recovery and Continuity of Operations
- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)
Similar topic areas

Security Program Management
  - Compliance Management
  - Threat Modeling
  - Risk Attitude
  - Product Supply Chain Risk Management
  - Services Third-Party Dependencies Management

Establishing and Maintaining Identities
  - Access Control
  - Asset, Change and Configuration Management
  - Physical Protection

Protection Model and Policy for Data
  - Implementation of Data Protection Controls

Vulnerability Assessment
  - Patch Management
  - Monitoring Practice
  - Situation Awareness and Information Sharing

Event Detection and Response Plan
  - Remediation, Recovery and Continuity of Operations

Access Control (AC)
  - Incident Response (IR)
  - Risk Management (RM)

Asset Management (AM)
  - Maintenance (MA)
  - Security Assessment (CA)

Awareness and Training (AT)
  - Media Protection (MP)
  - Situational Awareness (SA)

Audit and Accountability (AU)
  - Personnel Security (PS)

Configuration Management (CM)
  - Physical Protection (PE)

Identification and Authentication (IA)
  - Recovery (RE)

System and Communications Protection (SC)
  - System and Information Integrity (SI)
CMMC

- US Government based
- Guidelines for protection of government IP by vendors
- Certification of compliance
- Improvement between levels, reaching a higher level

SMM

- International
- For evaluating security of IoT solutions (sensors to the cloud including IT/OT/IoT)
- Certification of assessment companies that want to evaluate solutions
- Levels match need and investment goals, identify the right level
- Profiles and mappings
• Extending the scope to create an industry profile
  • Make general considerations more specific
  • Add more detail
  • Provide guidance on “what needs to be done”
  • Provide industry specific detail on “indicators of accomplishment”

• System-specific guidance can also be added for a system or a device
Relating SMM to familiar and accepted work

- SMM is a maturity model, so it does not include specific security controls
- Mappings take SMM “actions to be taken” and “indicators of accomplishment” for the maturity levels and relate them to control frameworks and best practices
- You can identify your desired maturity level for a given practice and appropriate controls that are relevant for it
Better together

• SMM Profiles
  – Extensions for different industries and purposes, for example
    • Retail
    • Digital Twins

• Mappings, for example
  – To other control security frameworks
  – ISA 62443 for different roles

• Could create SMM profile for CMMC to leverage SMM assessment approaches and mappings (such as for 62443) to identify gaps and possible controls for achieving CMMC levels

• Could combine SMM and CMMS certification review as part of an assessment, creates more consistency

• Call to action: let’s create an SMM CMMC profile
• SMM Main page: Security Maturity Model Practitioners’ Guide | Industry IoT Consortium (iiiconsortium.org)

• SMM White Paper: IoT SMM: Description and Intended Use (iiiconsortium.org)

• SMM Practitioner’s Guide: IoT SMM Practitioner’s Guide (iiiconsortium.org)

• SMM Retail Profile: IoT SMM: Retail Profile for Point-of-Sale Devices (iiiconsortium.org)
CMMC AND BSIMM

PRESENTED BY: Sammy Migues, Principal Scientist, Synopsys; BSIMM co-author and analyst

October 2021
CMMC and BSIMM

• What are they?
• What’s similar?
• What’s different?
• Complementary use
• Data
The **Cybersecurity Maturity Model Certification (CMMC)** is a new cybersecurity framework and accompanying certification by the US Department of Defense (DoD). The goal of the new CMMC compliance requirement is to protect Federal Contract Information (FCI) and Controlled Unclassified Information (CUI).

The **Building Security In Maturity Model (BSIMM)** is a descriptive, data-driven model resulting from an ongoing study—since 2008—of *actual practices* in application security programs across many organization sizes, industry verticals, and geographies. It’s both a yardstick for measuring programs and a guide for creating them.
### BSIMM Domains and Practices

#### GOVERNANCE
- Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.

#### INTELLIGENCE
- Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.

#### SSDL TOUCHPOINTS
- Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.

#### DEPLOYMENT
- Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.

#### PRACTICES

|----------------------------|----------------------|-------------------------------|-----------------------------|

### CMMC Domains

#### Domains
- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)
## Similar topic areas

### BSIMM Domains and Practices

<table>
<thead>
<tr>
<th>Domains</th>
<th>Practices</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Governance</strong></td>
<td>Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.</td>
</tr>
<tr>
<td><strong>Intelligence</strong></td>
<td>Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.</td>
</tr>
<tr>
<td><strong>SSD Ticketpoint</strong></td>
<td>Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.</td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td>Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.</td>
</tr>
</tbody>
</table>

### CMMC Domains

<table>
<thead>
<tr>
<th>Domains</th>
<th>Practices</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Access Control (AC)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Incident Response (IR)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Risk Management (RM)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Asset Management (AM)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Maintenance (MA)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Security Assessment (CA)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Awareness and Training (AT)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Media Protection (MP)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Situational Awareness (SA)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Audit and Accountability (AU)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Personnel Security (PS)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>System and Communications Protection (SC)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Configuration Management (CM)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Physical Protection (PE)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>System and Information Integrity (SI)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Identification and Authentication (IA)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Recovery (RE)</strong></td>
<td></td>
</tr>
</tbody>
</table>
# BSIMM Domains and Practices

## GOVERNANCE
- Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.

## INTELLIGENCE
- Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.

## SSDL TOUCHPOINTS
- Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.

## DEPLOYMENT
- Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.

## PRACTICES

<table>
<thead>
<tr>
<th>GOVERNANCE</th>
<th>INTELLIGENCE</th>
<th>SSDL TOUCHPOINTS</th>
<th>DEPLOYMENT</th>
</tr>
</thead>
</table>

## CMMC Domains

- **Access Control (AC)**
- **Incident Response (IR)**
- **Risk Management (RM)**
- **Asset Management (AM)**
- **Maintenance (MA)**
- **Security Assessment (CA)**
- **Awareness and Training (AT)**
- **Media Protection (MP)**
- **Situational Awareness (SA)**
- **Audit and Accountability (AU)**
- **Personnel Security (PS)**
- **System and Communications Protection (SC)**
- **Configuration Management (CM)**
- **Physical Protection (PE)**
- **System and Information Integrity (SI)**
- **Identification and Authentication (IA)**
- **Recovery (RE)**
## Similar topic areas

### BSIMM Domains and Practices

<table>
<thead>
<tr>
<th><strong>DOMAINS</strong></th>
<th><strong>GOVERNANCE</strong></th>
<th><strong>INTELLIGENCE</strong></th>
<th><strong>SSDL TOUCHPOINTS</strong></th>
<th><strong>DEPLOYMENT</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PRACTICES</strong></td>
<td><strong>GOVERNANCE</strong></td>
<td><strong>INTELLIGENCE</strong></td>
<td><strong>SSDL TOUCHPOINTS</strong></td>
<td><strong>DEPLOYMENT</strong></td>
</tr>
</tbody>
</table>

### CMMC Domains

- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)
## BSIMM Domains and Practices

<table>
<thead>
<tr>
<th>GOVERNANCE</th>
<th>INTELLIGENCE</th>
<th>SSDL TOUCHPOINTS</th>
<th>DEPLOYMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.</td>
<td>Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.</td>
<td>Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.</td>
<td>Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.</td>
</tr>
</tbody>
</table>

## CMMC Domains

- **Access Control (AC)**
- **Incident Response (IR)**
- **Risk Management (RM)**
- **Asset Management (AM)**
- **Maintenance (MA)**
- **Security Assessment (CA)**
- **Awareness and Training (AT)**
- **Media Protection (MP)**
- **Situational Awareness (SA)**
- **Audit and Accountability (AU)**
- **Personnel Security (PS)**
- **System and Communications Protection (SC)**
- **Configuration Management (CM)**
- **Physical Protection (PE)**
- **System and Information Integrity (SI)**
- **Identification and Authentication (IA)**
- **Recovery (RE)**
But they are different and have different context

**BSIMM**
- Any organization, anywhere
- Actual practices in current app security programs
- Program scorecard; self-assessments encouraged
- 122 unique activities / controls

**CMMC**
- DoD and defense contractors
- Cybersecurity best practices from various standards
- Certification element; self-assessments encouraged
- Level 3 includes 130 practices
### BSIMM Participants

<table>
<thead>
<tr>
<th>Category</th>
<th>BSIMM12</th>
<th>BSIMM11</th>
<th>BSIMM10</th>
<th>BSIMM9</th>
<th>BSIMM8</th>
<th>BSIMM7</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRMS</td>
<td>128</td>
<td>130</td>
<td>122</td>
<td>120</td>
<td>109</td>
<td>95</td>
</tr>
<tr>
<td>MEASUREMENTS</td>
<td>341</td>
<td>357</td>
<td>339</td>
<td>320</td>
<td>256</td>
<td>237</td>
</tr>
<tr>
<td>2ND MEASURES</td>
<td>31</td>
<td>32</td>
<td>50</td>
<td>42</td>
<td>36</td>
<td>30</td>
</tr>
<tr>
<td>3RD MEASURES</td>
<td>14</td>
<td>12</td>
<td>32</td>
<td>20</td>
<td>16</td>
<td>15</td>
</tr>
<tr>
<td>4TH MEASURES</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td>7</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>SSG MEMBERS</td>
<td>2,837</td>
<td>1,801</td>
<td>1,596</td>
<td>1,600</td>
<td>1,268</td>
<td>1,111</td>
</tr>
<tr>
<td>SATELLITE MEMBERS</td>
<td>6,448</td>
<td>6,656</td>
<td>6,298</td>
<td>6,291</td>
<td>3,501</td>
<td>3,595</td>
</tr>
<tr>
<td>DEVELOPERS</td>
<td>398,544</td>
<td>490,167</td>
<td>468,500</td>
<td>415,598</td>
<td>290,582</td>
<td>272,782</td>
</tr>
<tr>
<td>APPLICATIONS</td>
<td>153,519</td>
<td>176,269</td>
<td>173,233</td>
<td>135,881</td>
<td>94,802</td>
<td>87,244</td>
</tr>
<tr>
<td>AVG. SSG AGE (YEARS)</td>
<td>4.41</td>
<td>4.32</td>
<td>4.53</td>
<td>4.13</td>
<td>3.88</td>
<td>3.94</td>
</tr>
<tr>
<td>AVG. SSG RATIO</td>
<td>2.59/100</td>
<td>2.01/100</td>
<td>1.37/100</td>
<td>1.33/100</td>
<td>1.60/100</td>
<td>1.61/100</td>
</tr>
</tbody>
</table>

550 assessments across 231 firms since 2008
122 activities (controls) in BSIMM12
Trends seen in recent BSIMM data

• Governance automation and governance-as-code
• Continuous defect discovery
• Security as part of resilience and quality
• Growth in software supply chain risk management
• BSIMM Main page: Building Security In Maturity Model (bsimm.com)
PROGRAM BREAK

15-MINUTE BREAK

PROGRAM WILL RESUME AT 10:30AM EST

THANK YOU TO OUR SPONSORS FOR MAKING THIS EVENT POSSIBLE!
DEVOPS IMPLEMENTATION

PRESENTED BY:
Challenges in implementing and sustaining DevOps environment

Hasan Yasar
Director, Lifecycle Innovation and Automation
Software Engineering Institute | Carnegie Mellon University
Challenges in Implementing and sustaining DevOps environment

• DevOps Foundation
DevOps is a set of principles and practices which enable better communication and collaboration between relevant stakeholders for the purpose of specifying, developing, continuously improving, and operating software and systems products and services.

Four Fundamental Principles

1. **Collaboration**: between all stakeholders
2. **Infrastructure as code (IaC)**: assets are versioned, scripted, and shared
3. **Automation**: deployment, testing, provisioning, any manual or human-error-prone process
4. **Monitoring**: any metric in development or operation that can inform priorities, direction, and policy

* IEEE P2675 DevOps Standard for Building Reliable and Secure Systems Including Application Build, Package and Deployment
Dimension of DevOps

**Automation/Measurement**
- Automate repetitive and error-prone tasks (e.g., build, testing, and deployment) maintain consistent environments.
- Static analysis automation (architecture health).
- Performance dashboards.

**System Architecture**
- Architected to support test automation and continuous-integration goals.
- Applications that support changes without release (e.g., late binding).
- Scalable, secure, reliable, etc.

**Culture**
- All stakeholders collaborate.
- Developers and Operations support releases beyond deployment.
- Continuous learning.
- Transparent and sharable.
- Constant communication.

**Process and Practices**
- Pipeline streamlining.
- Continuous-delivery practices (e.g., continuous integration; test automation; script-driven, automated deployment; virtualized, self-service environments).
SW Development Phases – *on each iteration/sprint*
The DevOps Factory

- Feature to deployment
- Iterative and incremental development
- Automation in every phase of the SDLC
- Continuous feedback
- Metrics and measurement
- Complete engagement with all stakeholders
- Transparency and traceability across the lifecycle
Key Benefits of DevOps

- Reduced errors during deployment
- Reduced time to deploy and resolve discovered errors
- **Repeatable** steps
- **Continuous availability** of pipeline and application
- Increased innovation time
- **Responsiveness** to business needs
- **Traceability** throughout the application lifecycle
- Increased stability and quality
- **Continuous feedback**
# DevOps ROI

<table>
<thead>
<tr>
<th>Yearly Returns Possible from Cost of Unnecessary Rework Avoided</th>
<th>High IT Performer</th>
<th>Medium IT Performer</th>
<th>Low IT Performer</th>
</tr>
</thead>
<tbody>
<tr>
<td>LARGE ORGANIZATION that relies on in-house software (8,500 technical staff)</td>
<td>8,500 staff x $105,000 salary x 1.5 benefits x 1% rework = $13.4M</td>
<td>8,500 staff x $105,000 salary x 1.5 benefits x 12% rework = $160.7M</td>
<td>8,500 staff x $105,000 salary x 1.5 benefits x 7% rework = $93.7M</td>
</tr>
<tr>
<td>MEDIUM TO LARGE TECHNICAL ORGANIZATION (2,000 technical staff)</td>
<td>2,000 staff x $105,000 salary x 1.5 benefits x 1% rework = $3.2M</td>
<td>2,000 staff x $105,000 salary x 1.5 benefits x 12% rework = $37.8M</td>
<td>2,000 staff x $105,000 salary x 1.5 benefits x 7% rework = $22.1M</td>
</tr>
<tr>
<td>SMALL TO MEDIUM BUSINESSES AND NON-TECHNICAL ENTERPRISES (250 technical staff)</td>
<td>250 staff x $105,000 salary x 1.5 benefits x 1% rework = $393.8K</td>
<td>250 staff x $105,000 salary x 1.5 benefits x 12% rework = $4.7M</td>
<td>250 staff x $105,000 salary x 1.5 benefits x 7% rework = $2.8M</td>
</tr>
</tbody>
</table>

*DORA, DevOps ROI*
At what point in the development process does your organization perform automated application security analysis?

Mature DevOps practices are 338% more likely to integrate automated security.

350% 2019 survey
Challenges in Implementing and sustaining DevOps environment

• Obstacles & Recommendations
1. Culture
Incentivizing Behaviors

- Blame-Free Culture
  - No Hiding of Problems
  - Culture of shared responsibility
  - Collective decision and continuous learning

- Cross-Silo Goals
  - Incentivize Collaboration
  - Reduce “Not My Job”
  - Increase Sense of Purpose

- Optimize Ease-of-Use
  - Tools: Chat, ChatOps, Wiki
  - Integrated Pipelines
2. Organizational Structure
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 (like Security is everyone’s job)
- Keep team size small
3. Legacy Systems
Apply DevOps to migrate Legacy Systems

- Ancient systems should be replaced.
- Installing new systems to fit in
- Build a new version instead of maintaining
- Re-architect to support incremental and iterative development
- Enable dynamic integration of systems
# Tools complexity

<table>
<thead>
<tr>
<th>Periodic Table of DevOps Tools (V2)</th>
<th>Embed</th>
<th>Download</th>
<th>Add</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Tools complexity</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**XebiaLabs**

Enterprise DevOps

Follow @xebialabs

Publication Guidelines

---

Carnegie Mellon University

Software Engineering Institute

DevOps Environment

© 2019 Carnegie Mellon University

[DISTRIBUTION STATEMENT A] Approved for public release and unlimited distribution.
Platform as a Service for SW development

- 12Factor is a methodology for building software-as-a-service apps. [https://12factor.net](https://12factor.net)
- Heroko, [https://www.heroku.com](https://www.heroku.com)
- Cloud Foundry, [https://www.cloudfoundry.org](https://www.cloudfoundry.org)
- Pivotal, [https://pivotal.io](https://pivotal.io)
- Amazon DevOps, [https://aws.amazon.com/devops/](https://aws.amazon.com/devops/)
- OpenStack – Open Shift, [https://www.openshift.com](https://www.openshift.com)
- Electric Cloud [https://electric-cloud.com](https://electric-cloud.com)
- .......

[24x417]
Key Considerations

- Integrate-ability
- Interoperability
- Usability
- Portability
- Reliability
- Security/Permissions/ATO
- Availability
- Scalability

- Affordable
- Performance
- Modifiability
- Configurability
- “Automate-ability” (of manual tasks)
- “Approvability” (allows for manual approval)
- Measurability
- Adaptability
- Connectivity with other platform
DevOps Pipeline - General

- Communication Systems
- Issue Tracking System
- Build (CI) System
- Monitoring System
- Source Control
- Documentation System
- Code Review System
- Integration Environment
DevOps Pipeline with tooling
4. Lack of Metrics and Measurements
Decide what to measure

- Deployment frequency
- Change Lead time and Volume
- # of work items (tickets)
- Defect escape rate
- Mean time to detection (MTTD)
- Mean time to recovery (MTTR)
- Application performance
- Time to approval
- Time to patch vulnerabilities
- Operational Logs (IP, Stack Trace, Rate of Attack etc)
- Server/Services Usage (Disk, memory, CPU)
5. Process Challenges
DevOps Enabler..

Establish a process to enable people to succeed using the platform to develop Secure application

Such that;

• Constant communication and visible to all
• Ensures that tasks are testable and repeatable
• Frees up human experts to do challenging, creative work
• Allows tasks to be performed with minimal effort or cost
• Creates confidence in task success, after past repetitions
• Faster deployment, frequent quality release
6. DevOps and Acquisition
Apply DevOps Mindset

- Understand many portfolios of work as a continuous flow of smaller efforts

- Expand the collaboration, iteration, distributed (automated) governance constructs of Agile and DevOps to acquisition, needs analysis, certification, etc...
Compliance as Code

- Plan from beginning and carry-out throughout the lifecycle
- Enable audit log
- Design DevOps pipeline to comply with governance
- Make policy available to all stakeholders
- Implement configuration management and keep track every changes.
- Check and verify any configuration items
- Enable base configuration/OS
- Centralized and automated compliance policy
8. Inconsistent environments
Use Infrastructure as Code (IaC)

- Environment parity throughout the development pipeline
- Develop and treat provisioning scripts as part of code repository
- Share IaC amongst the developer and IT operational teams
9. Security (RMF, ATO)
6. **Monitor** the security controls in the information system on an ongoing basis including assessing control effectiveness, documenting changes to the system or its environment of operation, conducting security impact analyses of the associated changes, and reporting the security state of the system to designated organizational officials.

5. **Authorize** information system operation based on a determination of the risk to organizational operations and assets, individuals, other organizations, and the Nation resulting from the operation of the information system and the decision that this risk is acceptable.

4. **Assess** the security controls using appropriate assessment procedures to determine the extent to which the controls are implemented correctly, operating as intended, and producing the desired outcome with respect to meeting the security requirements for the system.

1. **Categorize** the information system and the information processed, stored, and transmitted by that system based on an impact analysis.

2. **Select** an initial set of baseline security controls for the information system based on the security categorization; tailoring and supplementing the security control baseline as needed based on an organizational assessment of risk and local conditions.

3. **Implement** the security controls and describe how the controls are employed within the information system and its environment of operation.
• **DevSecOps** is a model on integrating the software development and operational process considering security activities: requirements, design, coding, testing, delivery, deployment and incident response.
Think Security from Inception to Deploy and improve on every delivery
10. Sustaining
<table>
<thead>
<tr>
<th># Of Users</th>
<th># of App</th>
<th>Key CI/CD</th>
<th>Common Tools</th>
<th>Dev+Ops</th>
<th>Estimat ed Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;100</td>
<td>2-3</td>
<td>IaC, Single Cloud &amp; App Support, Monitoring and Notification</td>
<td>AWS, GCP, Docker, Jenkins, Nagios, PagerDuty, Slack, WikiPages, Jira, owasp zap, Twistlock</td>
<td>1 Dev + 1 Ops</td>
<td>$350K</td>
</tr>
<tr>
<td>100-1000</td>
<td>&gt;3, &lt;100</td>
<td>IaC, Hybrid Cloud &amp; App Support, Configuration Management, Env&amp;Pipeline Management, Release Strategies, Monitoring, Deployment Verification.</td>
<td>AWS, GCP, Docker, Jenkins, Nagios, PagerDuty, Slack, WikiPages, Jira, owasp zap, Twistlock, Circle CI, Sumo Logic, Logz.io, Kubernetes</td>
<td>3 Dev + 1 Ops</td>
<td>$750K</td>
</tr>
</tbody>
</table>

- Dev: $200K, Ops: $150K
- Platform depends on “on-prem or cloud”
How to sustain DevOps environment

• **Effective Usage**
  • Train Users and build DevOps skills
  • All stakeholders access
  • Playbook/Developers guidance
  • Project startup guidance
  • Project Architectural Guidance
    – Common Services,
    – Common Security approach
    – Architectural patterns
    – Test methods
• **DevOps environment usage policy**
  – Build and Deployment Strategies

• **Maintaining (cost/update)**
  • Updating the environment (new version or security patches)
  • Supporting new tools
  • Adding/setting up new projects
  • Operational Support
    – Base Image, OSS Support, Test harness, Temp Environment Creation
• **Pipeline orchestration**
• **Securing pipeline**
• **Usage meter/billing support**
• **Auditability/log and data collection**
SEI DevOps GitHub Projects

• Once Click DevOps deployment
  https://github.com/SLS-ALL/devops-microcosm

• Sample app with DevOps Process
  https://github.com/SLS-ALL/flask_api_sample
  • Tagged checkpoints
    • v0.1.0: base Flask project
    • v0.2.0: Vagrant development configuration
    • v0.3.0: Test environment and Fabric deployment
    • v0.4.0: Upstart services, external configuration files
    • v0.5.0: Production environment

• On YouTube:
  https://www.youtube.com/watch?v=5nQlJ-FWA5A
For more information...

DevOps: https://www.sei.cmu.edu/go/devops
DevOps Blog: https://insights.sei.cmu.edu/devops
Webinar : https://www.sei.cmu.edu/publications/webinars/index.cfm
Podcast : https://www.sei.cmu.edu/publications/podcasts/index.cfm
Hasan Yasar
Interim Director,
Lifecycle Innovation and Automation
hyasar@sei.cmu.edu
@securelifecycle
SOFTWARE SUPPLY
CHAIN TRANSPARENCY

PRESENTED BY:
SOFTWARE SUPPLY CHAIN TRANSPARENCY

Cross Cutting Solutions and Innovation Dept.
Cyber Solutions Innovation Center
MITRE Labs

9TH ANNUAL CYBER RESILIENCE SUMMIT
Virtual
October 12th, 2021
Software Supply Chain Integrity

Evidence Based Trust

Secure & Hardened Build and Distribution Infrastructure

Code → Commit → Build → Test → Package → Release → Deploy → Operate → Monitor

Producer

Consumer

Compare

Accept

SBOM

Jan 2021

Software Bill of Materials Standardization

ISO/IEC 5962:2021 SPDX®
OWASP CycloneDX
ISO/IEC 19770-2:2015 SWID tags

Open source components
Source files Developed by organization
Purchased components
Compilers
Libraries
Other documents: Multimedia, text
Build process
Makefiles
Generated code
Dynamic libraries
External executables
Binaries
Execution

Usage Scenarios Around SBOMs

Refer, Transfer or Purchase
(definition of what it is)

Pedigree
(history of how it was produced)

Provenance
(chain of custody of it)

Integrity
(cryptographic basis of unalteredness)

Proper and Legal
(conditions about its use)

Known Sw Vulns
(known fixes are applied to it)

Assurance
(safe-secure-resilient)

SBoM of a SW Service
(SBoM of sw delivering service)

Supply Chain Sequence Integrity

A non-profit service to improve the open source software supply chain by easing the adoption of cryptographic software signing, backed by transparency log technologies

**fulcio** – free Root-CA for code signing certs
- issues certificates based on an OIDC email address.
- only signs short-lived certificates valid for under 20 minutes.

**rekor** – the binary transparency log project under sigstore
- client CLI (for adding an entry to a rekor transparency log)
- pluggable PKI and support present for: GPG, X.509, Minisign

**cosign** – Container Signing, Verification and Storage in an OCI registry.
- aims to make signatures **invisible infrastructure**.
- supports: Hardware and KMS signing, Bring-your-own PKI, OIDC PKI (Fulcio), Built-in binary transparency and timestamping service (Rekor)
- Tested/demonstrated with the following registries:
  1. AWS Elastic Container Registry
  2. GCP's Artifact Registry and Container Registry
  3. Docker Hub
  4. Azure Container Registry
  5. JFrog Artifactory Container Registry
  6. The CNCF distribution/distribution Registry
  7. Gitlab Container Registry
  8. GitHub Container Registry
  9. The CNCF Harbor Registry
  10. Digital Ocean Container Registry
  11. Sonatype Nexus Container Registry
OCI Registry As Storage (ORAS)
https://github.com/oras-project

Tools and libraries to enable leveraging OCI registries for arbitrary artifacts

Open Container Initiative
https://github.com/opencontainers/

Creating open standards around container technology

OCI artifact manifest, Phase 1-Reference Types #29
The OCI artifact manifest generalizes the use of OCI image manifest, by reducing the constraints on all artifacts, enabling specific artifact-specs to set constraints for their type. Phase 1 adds support for artifacts to reference other artifacts through a subjectManifest property enabling reference graphs, as those required for secure supply chain efforts.

Phase 1: Reference Types
The PR focuses on Phase 1, enabling reference type support in 2021, supporting secure supply chain artifact types including signatures and SBoMs.

- OCI Artifacts Reference Types: github.com/opencontainers/artifacts/pull/29
- ORAS Reference Types: github.com/deislabs/oras/blob/reference-types/docs/artifact-manifest.md
- CNCF Distribution Reference Types:
github.com/notaryproject/distribution/blob/prototype-2/docs/reference-types.md
- Notary v2: github.com/notaryproject/notaryproject
Figure 1: Graphical depiction of the software supply chain with in-toto elements added. The project owner creates a layout with three steps, each of which will be performed by a functionary. Notice how the tag step creates `foo.c` and a localization file `foo.po`, which are fed to different steps down the chain.
Supply-chain Levels for Software Artifacts (SLSA)

SLSA guidelines have 4 levels of incremental and actionable things that software producers can claim to do to protect against specific integrity attacks.

https://github.com/slsa-framework/slsa
Supply Chain Integrity Model (SCIM)

Technologies leveraged:
- Attestations/Evidence, Confidential Ledgers, Hardware Roots of Trust, BOMs, CBOR (RFC 8949) and COSE (RFC 8152)

SCIM:
- defines minimum standards around the:
  - preparation, storage, distribution, consumption, validation and evaluation of arbitrary attestations/evidence about artifacts that are critical to maintaining the integrity of supply chains
  - specifies an end-to-end system for validating arbitrary artifacts in terms of supply chains whose integrity has been proven.
- is applicable to both hardware (objects in the physical world) and software (digital) artifacts.
- does not define how artifacts are produced or distributed, nor the methods by which attestations/evidence about artifacts are produced prior to preparation for inclusion in SCIM.
SCIM Usage Scenario

Ledger Store

web service

Supplier

Attester

User

Artifact 1 (d)

Evidence 1 (c)

Policy 1 (a)

Attestation 1 (e)

Policy 1 (e)

Artifact 1 (b)

Evidence 1 (c)
Trust Systems for a Supply Chain


Supply Chains – As multi-Stakeholder Network


© 2021 The MITRE Corporation. All rights reserved. Approved for Public Release; Distribution Unlimited. Case No: 21-01357-37
SOFTWARE SUPPLY CHAIN TRANSPARENCY

Robert Martin
Sr. Software and Supply Chain Assurance Principal Eng., MITRE

Allan Friedman
Senior Advisor & Strategist, CISA
KEYNOTE:

MODERNIZATION AND
DEVOPS BEST
PRACTICES AT AMAZON

PRESENTED BY:

LEO ZHADANOVSKY
CHIEF TECHNOLOGIST, US EDUCATION
AMAZON WEB SERVICES
Who am I?

• Worked for one of the biggest AWS customers in 2012
• 8 years at AWS
  • Ensure some of our biggest customer launches go smoothly
  • Help customers build modern applications
  • Work with customers and AWS service teams to meet our customer’s current and future needs
Today, we will cover

- Lessons learned at Amazon for developing and delivering software
- How to modernize your applications to take full advantage of the cloud
- How an AWS customer was able to scale successfully, despite extreme increases of demand caused by the pandemic
- Q&A
AWS in the public sector

- 7,500+ Government agencies
- 14,000+ Educational institutions
- 35,000+ Nonprofit organizations
Government, education, and nonprofit organizations are using AWS for digital transformation
We had three big ideas at Amazon that we have stuck with for 20+ years, and they are the reason we are successful: put the customer first, invent on our customers behalf, and be patient.

Jeffrey P. Bezos
Founder
Amazon.com, Inc.
Customer Case Study: Blackboard
Blackboard is a leading EdTech company, serving higher education, K–12, business, and government clients in every region of the world.

Blackboard connects a deep understanding of education with the power of technology to continuously push the boundaries of learning.

150M+ users in 80+ countries learn and communicate with Blackboard tools.
Blackboard virtual classrooms
Adapting to unpredictable spikes in traffic

• March – Start of school year in southern hemisphere
• March 7th, 2020
  • “We must stop, contain, control, delay and reduce the impact of this virus at every opportunity.” – World Heath Organization
• 4800% increase in usage on Collaborate compared to pre-pandemic
“We had entire countries shifting to online learning overnight ... not only did we have to accommodate the increased usage, but we also had to support the institutions as they shifted their entire paradigm from onsite to online learning.”

Kris Stokking
VP of Software Engineering
Blackboard
How did Blackboard scale?

- At first, overprovision
- Next – autoscaling
- Diversify compute
- Let AWS handle undifferentiated heavy lifting
- Partner with AWS
  - Support
  - Solutions Architecture
- Learn more here: https://aws.amazon.com/solutions/case-studies/blackboard-ec2-case-study
“AWS provides value in a way that empowers Blackboard to really focus on its core value proposition ... Blackboard is more agile and better equipped to deal with change because we’re on AWS.”

Kris Stokking
VP of Software Engineering
Blackboard
Amazon’s Modernization and DevOps Journey
One area where I think we are especially distinctive is failure. I believe we are the best place in the world to fail (we have plenty of practice!), and failure and invention are inseparable twins. To invent you have to experiment, and if you know in advance that it’s going to work, it’s not an experiment.

Jeffrey P. Bezos
Founder
Amazon.com, Inc.
Lessons learned at Amazon for developing and delivering software

- Decompose for agility *(microservices, 2-pizza teams)*
- Automate everything
- Standardized tools
- Belts and suspenders *(governance, templates)*
- Infrastructure as code
Just starting out

This is how many web architectures started out, and it’s how Amazon started too...

There any many bottlenecks, and scaling of the web server was an immediate factor
Going further

Principles
• Make units a small as possible (Primitives)
• De-couple based on scaling factors, not functions
• Each service operates independently
  “Communication is terrible!” — Jeff Bezos
• APIs (contracts) between services
Impact to our development
Monolith development lifecycle
Monolith development lifecycle

This led to changes in organization
Impact to our organization
Getting (re)organized

“Two-pizza” teams

- Own a service
- Minimizes social constraints (Conway’s law)
- Autonomy to make decisions
Transformation timeline

2001

Monolithic application + teams

2002

Microservices + 2-pizza teams
Teams Own Everything

- Planning
- Security
- Performance
- Scalability
- Deployment

- Operation
- Bugs
- Documentation
- Testing...
Now we have...
Modern applications

Today we have modern applications

- Use independently scalable microservices (serverless, containers...)
- Connect through APIs
- Deliver updates continuously
- Adapt quickly to change
- Scale globally
- Are fault tolerant
- Carefully manage state and persistence
- Have security built-in
Modernization

“How do I get to modern applications?”
Migration and Modernization Patterns

AVERAGE CUSTOMER ENVIRONMENT, BY MIGRATION PATTERN
(based on AWS experience)

<table>
<thead>
<tr>
<th>Migration Pattern</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REHOST</td>
<td>Lift-and-shift to quickly capture business benefits, transform once migrated for additional value</td>
</tr>
<tr>
<td>RELOCATE</td>
<td>Move VMWare-based apps to VMWare Cloud on AWS (VMC)</td>
</tr>
<tr>
<td>REPLATFORM</td>
<td>Platform optimizations (e.g. to reduce OS/DB licensing costs)</td>
</tr>
<tr>
<td>REFACTORS</td>
<td>Reimagine app architecture &amp; development</td>
</tr>
<tr>
<td>REPURCHASE</td>
<td>Move to SaaS</td>
</tr>
</tbody>
</table>

Full spectrum of patterns is important for transformation – but up to ~60% of typical environment can be rapidly migrated at a predictable price, freeing time & budget to focus on modernization
Modernization Pathways

- **Move to Cloud Native Architecture**
  - Agile, scalable apps built on containers, serverless and microservices

- **Move to Managed Cloud Services**
  - Deploy applications rapidly and operate reliably at scale with managed services

- **Move to Managed Databases**
  - Open source, fit for purpose, highly scalable databases

- **Move to Open Source**
  - Freedom from proprietary licensed software with open source technology
Martin Fowler’s Strangler Pattern

“...gradually create a new system around the edges of the old, letting it grow slowly over several years until the old system is strangled.”

Martin Fowler
June 29, 2004
## Success Stories

<table>
<thead>
<tr>
<th>Company</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edmunds</td>
<td>Built an image processing solution in just days using AWS serverless. The solution processed 50 million images in 8 days for $6,000 and is now saving more than $100,000 per year.</td>
</tr>
<tr>
<td>FINRA</td>
<td>Re-architected on-premises Hadoop cluster to AWS serverless in 3 months; increased cost efficiency by 2x while handling half a trillion stock trade validations a day, improving security and compliance.</td>
</tr>
<tr>
<td>Bustle</td>
<td>Modernized and built an entirely serverless website with half the team size normally required to build and operate at their scale. Freed up engineers to focus on building out new features and innovating, and realized 84% cost savings.</td>
</tr>
<tr>
<td>Mitsui Coal</td>
<td>Improved system uptime from 85% to 99.99% after migrating its finance reporting system from on-premises to SAP S/4HANA on AWS.</td>
</tr>
<tr>
<td>BP</td>
<td>Increased SAP ERP response times by 40% while lowering costs to support supply chain initiatives.</td>
</tr>
</tbody>
</table>
Thank you!

Leo Zhadanovsky
Our mission

- Our task was to improve:
  - Innovation
  - Speed
  - Agility
  - Safety

- What we did:
  - Decomposed for agility
  - Cultural and operational shift
  - Created tools for software delivery
Scaling v1

In 1998 the “Distributed Computing Manifesto” came out and we began breaking things down into separate components...

This was a bit better, still not very scalable
TESTING FOR DATA PRIVACY AND PROTECTION

PRESENTED BY:
ENSURING SECURE & RESILIENT IT MODERNIZATION OUTCOMES

PRESENTED BY:
DAVID POWNER, EXECUTIVE DIRECTOR CENTER FOR DATA-DRIVEN POLICY, MITRE

SETH CARMODY
Why IT Modernization is Important

- Strengthens security posture
- Reduces technical debt
- Advances mission outcomes and citizen services
- Positions us for future scalability and maintainability
Recent History Calling for Attention

2016-2017
• 2016: GAO releases report on modernizing IT legacy systems
• 2017: Report to the President on Modernization
• 2017: NDAA MGT Act

2018-2020
• 2018: President’s Management Agenda (3 Priorities)
• 2019: GAO releases their second report
• 2020: Senator Hassan questions to 10 federal agencies

2021
• Senate Hearings
• PMA?
• Legacy Reduction Act Legislation?
Focus and Results to Date

- Business System modernization
- Cloud Migration
- Limited Mission Critical Legacy System Modernization
  - Wartime readiness
  - Tax Processing
  - Benefit programs
Path Forward

• Move beyond just identifying Legacy Systems

• Prioritize – criteria/MITRE recent research

• Plans/transparency/accurate budgets

• Progress against plans
  • Business ownership/SW SCRM/ISO 5055/CX

• OMB help
Why Healthcare Cybersecurity is Hard

Seth Carmody, PhD
VP Regulatory Strategy

CISQ Cyber Resilience Summit
October 12, 2021
INCREASED TECHNICAL DEBT

Consumers are responsible for managing security debt passed to them from producers.

No investment

- Tech
- MDM
- HDO
- Clinicians
- Patients

REGULATORS: FDO & Congress
May cyberattack cost Scripps nearly $113M in lost revenue, more costs

by Robert King | Aug 11, 2021 3:55pm

Global Edition  Privacy & Security

Nevada hospital ransomware attack could affect data of 1.3M patients

An Ohio-based law firm is investigating claims on behalf of the breach victims.

By Kat Jerchich | August 23, 2021 | 05:02 PM
Computers Aren’t Pills

“We Need to Rethink the Whole Thing”
Jeff Shuren Will Aim High on MDUFA V-Linked Reforms

Computers == Pills
=> FALSE
INCREASED SECURITY INVESTMENT

Initial security investment increases security posture and reduces costs for consumers.

+Investment  Tech  MDM  HDO  Clinicians  Patients

REGULATORS: FDO & Congress
Questions?

https://medcrypt.com/whitepapers-medical-device-thoughtleadership.html

seth@medcrypt.co
SUMMARY AND CLOSING REMARKS

PRESENTED BY:
DR. BILL CURTIS
LUKE MCCORMACK
THANK YOU FOR ATTENDING!