James R. McQuiggan, CISSP is the Product & Solutions Security Officer of the Service Americas division at Siemens Gamesa Renewable Energy. He is responsible for developing the strategy and policies for securing the SCADA products deployed to America’s customers. With over 17 years at Siemens, James has consulted and supported various corporate divisions on cybersecurity standards, information security awareness and securing their networks.

James is an ISSA member and the Past President of the Central Florida chapter from 2012 to 2016. He supported the ISSA International Conference as the Content Committee Chair from 2014 to 2017. He is a member of the Central Florida chapters of ISACA, Infragard and (ISC)2. Within (ISC)2, he is a member of the Advisory Council of North America and is the (ISC)2 Central Florida chapter President. James is the recipient of the 2016 (ISC)2 President’s Award and in 2017 he was awarded the first ever Julie Peeler Franz “Do it for the Children Award” as Volunteer of the Year for his work with the Safe & Secure Online Education program, that provides resources to help children, parents, educators and senior citizens stay safe online.

Most importantly, James is the father of two teenage daughters and continues to make sure they are safe and secure online while using their smartphones and social media.
Derek E. Weeks is the world's foremost researcher on the topic of DevSecOps and securing software supply chains. For the past five years, he has championed the research of the annual State of the Software Supply Chain Report and the DevSecOps Community Survey. Derek is a huge advocate of applying proven supply chain management principles into DevOps practices to improve efficiencies and sustain long-lasting competitive advantages. He is a frequent keynote speaker and industry panelist at conferences like the RSA Conference, Bosch IoT Summit, Red Hat Government Summit, Carnegie Mellon's SEI Software Supply Chain Summit, and numerous DevOps Days events. He currently serves as vice president and DevOps advocate at Sonatype, creators of the Nexus repository manager and the global leader in solutions for software supply chain automation. Derek is also the co-founder of All Day DevOps, an online community of 65,000 IT professionals. In 2018, Derek was recognized by DevOps.com as the "Best DevOps Evangelist" for his work in the community.
47%

deploy multiple times per week

Source: 2019 DevSecOps Community Survey

velocity
59,000 data breaches
have been reported to GDPR regulators since May 2018

Source: DLA Piper, February 2019
Everyone has a software supply chain.
(even if you don't call it that)
Demand drives 15,000 new releases every day

Source: Sonatype

@weekstweets
Automation accelerates OSS downloads

Download Requests for Java Components 2008-2018 are a proxy for the popularity of automated software development.

Source: Sonatype’s 2018 State of the Software Supply Chain Report
ROLLING WEEKLY DOWNLOADS OF NPM PACKAGES

Sources: npm Inc., Laurie Vess (Arsedio)
85% of your code is sourced from external suppliers
170,000 Java component downloads annually

3,500 unique

Source: 2018 State of the Software Supply Chain Report
60,660
JavaScript packages
downloaded per developer
per year

source: npm, 2018
Not all parts are created equal.
We are not “building quality in”.

12.1%

2017
Java

NOT REFLECTIVE OF THE HARTFORD’S DATA
source: 2019 State of the Software Supply Chain Report
We are not “building quality in”.

51%

2018

source: 2018 npm
170,000 java component downloads annually

3,500 unique

18,870 11.1% with known vulnerabilities
60,660
JavaScript packages downloaded annually per developer

30,330
51% with known vulnerabilities
Developers continue to believe security is important but *don’t have enough time to spend on it.*

- 50% in 2017
- 48% in 2018
- 48% in 2019
Breaches increased 71%

source: DevSecOps Community Survey 2014 and 2019
DevSecOps challenge: automate faster than evil.

Sources: Gartner, IBM, Sonatype
Quickly identify who is faster than their adversaries

47% deploy multiple times a week

Source: 2019 DevSecOps Community Survey
Equifax was not alone

March 7
Apache Struts releases updated version to thwart vulnerability CVE-2017-5638

March 8
NSA reveals Pentagon servers scanned by nation-states for vulnerable Struts instances

March 9
Cisco observes "a high number of exploitation events."

March 10
Equifax
Canada Revenue Agency
Canada Statistics
GMO Payment Gateway

March 13
Okinawa Power
Japan Post

The Rest of the Story

March 10
Equifax

March 13
Okinawa Power
Japan Post

March 18
India’s AADHAAR

March 9
Cisco observes "a high number of exploitation events."

March 8
NSA reveals Pentagon servers scanned by nation-states for vulnerable Struts instances

Struts exploit published to Exploit-DB.

The Rest of the Story

December ’17
Monero Crypto Mining

Today
65% of the Fortune 100 download vulnerable versions

@weekstweets
Complete software bill of materials (SBOM)

Source: 2019 DevSecOps Community Survey
The new battlefront
Software Supply Chain Attacks

0. July 2017
Homebrew repository compromised.

1. Aug 2017
npm event-stream attack on CoPay.

2. Sep 2017
10 Malicious Python packages
Basic info collected and sent to Chinese IP address

3. Oct 2017
Blog: “I’m harvesting credit card numbers and passwords from your site. Here’s how.”

4. Nov 2017
Golang go-bindata github id deleted and reclaimed.

5. Dec 2017
Conventional-changelog compromised and turned into a Monero miner.

6. Jan 2018
Backdoor discovered in npm get-cookies module published since March.
Unauthorized publishing of mailparser.

7. Feb 2018
ssh-decorator Python Module stealing private ssh keys.

8. Mar 2018
Gentoo Linux Repository Compromised.

9. Apr 2018
Malicious Eslint discovered to be stealing npm credentials.

10. May 2018

11. Jun 2018

12. Jul 2018

13. Aug 2018

14. Sep 2018

15. Oct 2018

16. Nov 2018

17. Dec 2018

Study found credentials online affecting publishing access to 14% of npm repository. +79,000 packages.
Malicious npm Packages “typosquatted” (40 packages for 2 weeks. Collecting env including npm publishing credentials).

Image by Sonatype
At what point in the development process does your organization perform automated application analysis?

- **Design / Architecture**: 7% (2019 No DevSecOps Practice), 22% (2019 Mature DevSecOps Practices)
- **Development**: 17% (2019 No DevSecOps Practice), 44% (2019 Mature DevSecOps Practices)
- **Build/CI**: 38% (2019 No DevSecOps Practice), 74% (2019 Mature DevSecOps Practices)
- **During QA / Test**: 20% (2019 No DevSecOps Practice), 55% (2019 Mature DevSecOps Practices)
- **Prior to release into production**: 18% (2019 No DevSecOps Practice), 51% (2019 Mature DevSecOps Practices)
- **Production**: 19% (2019 No DevSecOps Practice), 53% (2019 Mature DevSecOps Practices)
- **Throughout the process**: 10% (2019 No DevSecOps Practice), 45% (2019 Mature DevSecOps Practices)

Mature DevOps practices are 350% more likely to integrate automated security.
Which application security tools are used?

<table>
<thead>
<tr>
<th>Tool</th>
<th>2019 No DevSecOps Practice</th>
<th>2019 Mature DevSecOps Practices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Application Firewall</td>
<td>70%</td>
<td>85%</td>
</tr>
<tr>
<td>Container and Application Security</td>
<td>60%</td>
<td>91%</td>
</tr>
<tr>
<td>Open source governance</td>
<td>56%</td>
<td>84%</td>
</tr>
<tr>
<td>Static Application Analysis</td>
<td>55%</td>
<td>77%</td>
</tr>
<tr>
<td>Dynamic Application Analysis</td>
<td>40%</td>
<td>70%</td>
</tr>
</tbody>
</table>
How are you informed of InfoSec and AppSec issues?

Automating security enables faster DevOps feedback loops.
Automation continues to prove difficult to ignore

2019 No DevOps Practice

- 75% no policy or ignore it
- 25% YES

2019 Mature DevOps Practices

- 62% YES
- 38% no policy or ignore it

Source: 2019 DevSecOps Community Survey
Trusted software supply chains are 2x more secure

Source: 2018 State of the Software Supply Chain Report
The rising tide of regulation and software liability
1. An up to date inventory of open-source components utilized in the software

2. A process for identifying known vulnerabilities within open source components

3. 360 degree monitoring of open source components throughout the SDLC

4. A policy and process to immediately remediate vulnerabilities as they become known

“Emphasize performance of the entire system and never pass a defect downstream.”
QUESTIONS?
Matthew Small leads Solution Architecture for Bitnami. He is passionate about the cultural values and technology that enables teams to operate successfully in a cloud-native paradigm. To that end, he works with their hyperscale cloud provider partners as well as enterprise customers to leverage and integrate Bitnami's DevSecOps products and certified Open Source Software. Prior to joining Bitnami, Matthew spent 10 years at RightScale, a pioneer in multi-cloud and hybrid-cloud management and thought leader in cloud best practices. He has his head stuck in the clouds.
Introducing Bitnami

We make awesome software available to everyone, everywhere.

Company Profile:

- Founded in 2005
- Self Funded and Profitable
- HQ in San Francisco
- 75+ Employees Globally
- Strategic Partner to most Clouds

Cloud Product Portfolio:

- OSS Adoption (Free App Catalog)
- Enterprise OSS (Enterprise Catalog)
- App Packaging / DevSecOps (Stacksmith)
- Kubernetes (Helm, Kubeapps, BKPR)

About Matthew Small:

Head of Solutions Architecture, Enterprise and Cloud Platforms

Over 11 years in Cloud

Focused on:

- Enterprise Cloud Strategy
- DevSecOps Automation
- Multi/Hybrid-Cloud Environments
- Distributed Architectures
- Container Enablement

Company Profile:

- Founded in 2005
- Self Funded and Profitable
- HQ in San Francisco
- 75+ Employees Globally
- Strategic Partner to most Clouds

Cloud Product Portfolio:

- OSS Adoption (Free App Catalog)
- Enterprise OSS (Enterprise Catalog)
- App Packaging / DevSecOps (Stacksmith)
- Kubernetes (Helm, Kubeapps, BKPR)
Agenda and Contents

Our Software Defined World
The State of Cloud Security
Examining 3 Current Trends
  DevSecOps
  Immutability
  Application Packaging
Our Software Defined World

Everything is Defined as Code
- Applications
- Configurations
- Security operations
- Infrastructure and services
- Deployments

Code and Components are Modular
- Sourced from Public and Private Repositories
- Sourced from 3rd Parties
- Leveraged as PaaS
- Defined as FaaS
- IaC libraries and documentation snippets
- Stack Overflow snippets

Pipelines and Processes must Evolve
- Defining Trust Circles
- Application Packaging
- Governance and Guardrails
- Continuous Everything
The State of Cloud Security

According to a recent survey from Oracle and KPMG…

Source: [https://www.oracle.com/cloud/cloud-threat-report/](https://www.oracle.com/cloud/cloud-threat-report/)

92% Are concerned that security policies around the use of cloud applications and services are being violated

82% Have experienced security events due to confusion over Shared Responsibility Security Models

69% Are aware that unapproved cloud applications are being used moderately to significantly

50% Report that unsanctioned uses of cloud has resulted in unauthorized access (and malware in 48% of those cases)

23% Don’t have the human capital resources to keep up with system patches and maintenance

3 Key Takeaways

1. Cloud security in the enterprise is immature relative to cloud adoption.
2. Unapproved processes and privileges, and general misunderstandings cause tangible problems.
3. Humans can’t keep up, and automation of security processes is the only way to stay ahead.
Bitnami
Examining Current Trends
Trend: DevSecOps

Dev is:
Managing application code collaboratively
example: source control and continuous integration

DevOps is:
- Managing infrastructure as code collaboratively
  example: deployment templates and continuous deployment

DevSecOps is:
- Enforcing security along the way
  example: continuous packaging
### Enabling Customer Centric and Cloud-ready IT Operations

<table>
<thead>
<tr>
<th><strong>Traditional SecOps</strong></th>
<th><strong>DevSecOps</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology focused</td>
<td>Business value focused</td>
</tr>
<tr>
<td>Dictates technology standards</td>
<td>Consults/collaborates on requirements</td>
</tr>
<tr>
<td>Narrow operations focus</td>
<td>Focused on customer experience</td>
</tr>
<tr>
<td>Heavy handed implementation</td>
<td>Partner and “shift left”</td>
</tr>
<tr>
<td>Focused on internal solutions</td>
<td>Leverages external sources and vendors</td>
</tr>
<tr>
<td>Adverse to change</td>
<td>Embraces iterative change</td>
</tr>
<tr>
<td>“No”</td>
<td>“Here’s what’s possible”</td>
</tr>
</tbody>
</table>
Establishing Trusted Sources and a Shared Security Model

- **Community Sources**
  - **Risk:** Uncontrolled, unmaintained, unsupported, dependency web
  - **Reward:** Open source, flexible, highly modular
  - **Tip:** Compile from source and scan absolutely everything

- **Cloud Provider Sources**
  - **Risk:** Cloud specific implementations (lock in)
  - **Reward:** Predefined best practices, works out of the box
  - **Tip:** Leverage reference architectures in your own automation

- **Certified Sources**
  - **Risk:** Limited selection of components and applications
  - **Reward:** Pre-tested, pre-scanned, available across clouds
  - **Tip:** If it’s undifferentiated, then use a certified source (OS, OSS).

- **Private Sources**
  - **Risk:** Depends on enforcement of upstream practices
  - **Reward:** Everything in a trusted repo can be trusted by the org
  - **Tip:** Distill everything down and proxy through a private source
Addressing the 3 Biggest Challenges in DevSecOps Adoption

Frictionless Compliance at DevOps Speed

How can you inject security requirements without major impact on the day to day work?

*Apply preventative measures at build and deploy time and enable scanning, detection and reactive automation for operational policies.*

Managing the Frankenstack

And still provide tool flexibility and choice to individual Dev, Sec and Ops teams?

*Don’t try to standardize on tools in a large org, instead standardize on process.*

Updating Security Policies

And how do you enforce a changes at scale wherever they are needed?

*Centralized build automation can ensure changes are propagated.*
## Trend: Immutability

Immutable is: **Unchanged once Deployed**  
(and the only way to do containers)

<table>
<thead>
<tr>
<th>Benefits to Security</th>
<th>Challenges to Adoption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less vectors of attack</td>
<td>Requires process change</td>
</tr>
<tr>
<td>Environment consistency (dev/test/prod)</td>
<td>Artifact dependency web</td>
</tr>
<tr>
<td>Never log into a server again</td>
<td>Patches must be included in artifacts</td>
</tr>
<tr>
<td>Simplifies and accelerates deployments</td>
<td>Automation is crucial for max benefit</td>
</tr>
<tr>
<td>Eliminates configuration drift</td>
<td>Configurations must be versioned</td>
</tr>
<tr>
<td>Easy to roll back</td>
<td>Small changes require new deploys</td>
</tr>
<tr>
<td>Defined point in time releases</td>
<td>Artifacts age and must be regenerated</td>
</tr>
</tbody>
</table>
The Immutability Staircase

From baseline OS images up to full containers.

- **Example**: Golden OS Image
- **Example**: Chef Baseline
- **Example**: .Net Core Baseline
- **Example**: A Container
The Challenge and Opportunity with Containers

The Challenge: Containers force you into a new operating model.

Data Center: Mutable, Repair vs Replace
Cloud: Mutable or Immutable, Repair or Replace
Containers: Immutable Only, Replace vs Repair

The Opportunity: Those models can be applied to legacy operations.
An Application Package is:
The artifacts and automation required to deliver a business outcome.
What’s in an App Package?

An Application Package is: **Encapsulated by a Deployment Template**

**Examples:**
- AWS Cloud Formation Templates
- Azure ARM Templates
- Terraform Templates
- VMware Blueprints
- ServiceNow Blueprints
- Kubernetes Helm Chart
- CNAB (Cloud Native Application Bundle)

**Not Examples:**
- Configuration Management Scripts

Tip: Treat your deployment templates as immutable and version them.
How Most Organizations Operate Today without App Packaging

A well meaning mess of best efforts and meetings...

Build Software
- Manage Source
- Integrate Code
- Unit Test
- External Sources

Generate/Maintain Artifacts
- DevOps Release Management
- Meetings
- Spreadsheets
- Tribal knowledge
- Security IT Operations

Deploy Applications
- Canary & Testing
- New Release
- Update Existing
How Organizations Operate with Application Packaging

A codified, programmatically enforced, continuous process...

Build Software
- Manage Source
- Integrate Code
- Focuses on the Application: Code, Dependencies, Boot/Run Logic
- Unit Test

Generate/Maintain Artifacts
- Generate Images
- Converges the Best Practices: Images, Automation, Isa5/Psa5/Fae5
- Enforce Security Configurations

Deploy Applications
- Author Templates
- Canary & Testing
- Measures the Impact and SLAs: Uptime, Performance
- New Release
- Update Existing

Manage System Packages
- Updates
- CVEs
Evaluating or implementing Enterprise DevSecOps, Kubernetes and Open Source Software use and strategies?

Contact us at: enterprise@bitnami.com
Check out our DevSecOps framework: bitnami.com/stacksmith
Start with our free open source apps: bitnami.com/stacks
Learn more about our Enterprise OSS Catalog: bitnami.com/enterprise-catalog

Getting Started with Kubernetes?

Grab Essential Pre-integrated Tools: bitnami.com/projects/kubeapps-and-bkpr

Just want to chat?

Email: mattsamall@bitnami.com
LinkedIn: in/mattasmall
Twitter: @matthewsmall
QUESTIONS?
Jimmy Rabon serves as a senior product manager for Fortify products at Micro Focus. In this role, he is responsible for defining the strategy, roadmap, and features for Fortify Software Security Center, Tools, and the Fortify ecosystem with the goal of making software more secure.

Jimmy Rabon began his career as an application developer for the Computer Sciences Corporation developing in a variety of programming languages and frameworks before deciding to specialize in application security, joining Fortify software. An 8-year veteran of the application security industry, Jimmy began his career in this field by serving as a software security consultant for Fortify software. He has performed countless security audits (static and dynamic) of applications with a very large range of programming languages and technology stacks for both commercial and government enterprises. He has helped customers worldwide to understand and prioritize their security risks and provided the expertise necessary to build world class software assurance programs.

Having worked as a software engineer for many years prior to specializing in application security, he understands the unique challenges that developers face when attempting to deliver secure code and can help deliver the effective technology and processes that enable effective application security at scale and speed.

He held positions at Fortify of Software Security Consultant, Managing Consultant, Fortify Technical Director, and Fortify Services Director. He led a team of highly skilled application security consultants as the head of HPE Fortify professional services.

Jimmy earned a bachelor’s degree in Management of Information Systems from Auburn University and is a Certified Secure Software Lifecycle Professional.
Enterprise DevOps by Design

Continuous Integration & Testing
Continuous Delivery & Deployment
Continuous Operations
Continuous Assessment

Increase Automation
Reduce Latency
Increase Visibility
App Security Challenges

➢ Smaller more frequent releases
  ○ Automation is an undeniable right

➢ Smaller cross functional teams can potentially complicate integration of security tooling / processes into the SDLC with inconsistent workflows or tech stacks

➢ Failing early must include security for this to work

➢ Shift Left Approach Adjusts Traditional Personas

➢ Singular effective communication channel regardless of the tool or contributor
Training

➢ Lead with training: What is finding if the person responsible isn’t confident in the finding or the fix??

➢ Self-service application security models must include contextually correct training content delivered efficiently (Automation)

➢ Development best practices and guidelines should be incorporated into these models as organizations mature
Static analysis is less challenging to automate the scan but what about auditing??

- CLI / Build Tools / Systems / Webhooks / APIs
- Treating builds as code has advantages for static integration (Pipelines)
- Self-service model should strive to automate onboarding, integration, and communication
- Auditing is the most time consuming aspect
  - Leverage historical audit data with machine learning to provide audits for future results

Dynamic analysis has evolved into small automated scans driven through QA artifacts to efficiently exercise specific tests in a timely manner
Flexibility

➢ All security tooling (whether static or dynamic analysis) must expose complete functionality of the platform through well documented APIs
  ❑ Samples of common workflows achieved through API usage can “enable” any DevOp workflow
  ❑ Say yes to a workflow instead of security blocking development progress
  ❑ Ability to import any type of data into vulnerability management platforms
  ❑ Ability to efficiently export data into centralized data management platforms
The Shift Left Approach

- Change in the separation of duties from IT operations and development
- Continuous security testing must be the early warning detection system
  - Certain classes of vulnerabilities can be found with very efficient purpose built analyzers and include real-time detection for increased efficiency (IDE or outside)
  - Build instability / failure CAN be effective if narrowed down to a very specific criteria based on really good data
  - Identification of trusted data sources is often possible early with increased developer role in security responsibilities
The most important topic to consider with regards to modern DevOps

Question: What is the most effective means to communicate a security vulnerability to a developer?

Answer: The answer to that question tells you all you need to know

- ChatOPs
- Bug Tracking
- Ticket Management
- Pigeon

Advanced two way integrations from security platforms to other platforms are often needed

Application security is another data feed into the executive dashboard
QUESTIONS?
Bjoern is the CEO and co-founder of Templarbit, a security company that helps businesses collect and use security intelligence to mitigate risk on web apps and APIs. Templarbit was part of Y Combinator’s summer 2017 batch. Prior to Templarbit, Bjoern led engineering at Synack, where he helped protect the Pentagon, DoD, IRS and Fortune 100 clients from security breaches.
The future of AppSec is data driven
BJOERN ZINSSMEISTER
CEO at Templarbit
Keeping a software business secure and risk free is getting much harder

system_complexity++;
development_speed++;
It has never been easier to deploy code.
Faster development cycles lead to more security implications
AppSec Tools

Static Code Analysis

Web Application Firewall
Basic check
Not available for all languages
Often used to highlight part of the code base for further manual investigation
Web Application Firewall

Filters bad traffic

Often can be bypassed

Hard to configure and maintain, often introducing more ongoing work
20+ year old tools
don’t work at 2019 scale
A data driven approach to AppSec will be a better future
Collect Security Intelligence and make it useful
AppSec Intelligence

Monitor shape of the application

Detects new features

Highlight areas that are sensitive
Battle-tested Core Rules

WAF core and custom rules

Executed directly on app server

Static code analysis for pre-prod signals
Machine Learning

Automate work with ML where you can

Detect anomalies in the data set

Global and local models for maximum coverage
What this looks like out in the field
Detecting malicious traffic

No signs of WAF rule violation, but ML model classifies behavior as malicious

Connection is blocked
Finding areas of the App that need a security review

March 2018: A new feature was released

AppSec Intelligence change in app shape to security team

December 2018: Feature is changed

AppSec Intelligence detects change and highlights this for the security team

December 2018: Security completes review

Feature is marked as reviewed
Thank you

www.templarbit.com
QUESTIONS?