

Security Practitioner Perspective on DevOps for Building Secure Solutions

Zane Lackey

Hasan Yasar

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Copyright 2016 Carnegie Mellon University

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

Carnegie Mellon® and CERT® are registered marks of Carnegie Mellon University.

DM-0004111



This talk will cover the perspectives of security practitioners on building secure software using the DevOps development process and modern security approach.





Building Secure Solutions
DevOps Foundations

The DevOps Movement Began as a Reaction ...



to years of disconnect between Development and Operations that began to manifest itself as conflict and inefficiency



What is DevOps?

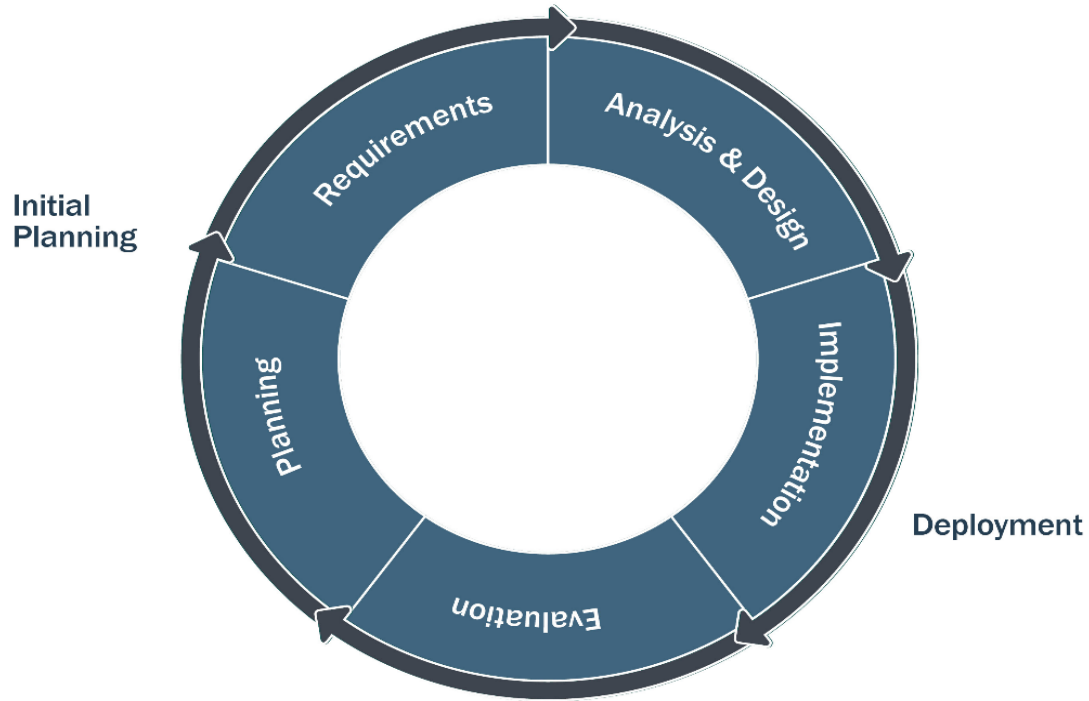
DevOps (a portmanteau of "development" and "operations") emphasizes *communication*, *collaboration*, and *integration* between software developers and information technology (IT) operations personnel. [1]

[1] <http://en.wikipedia.org/wiki/DevOps>



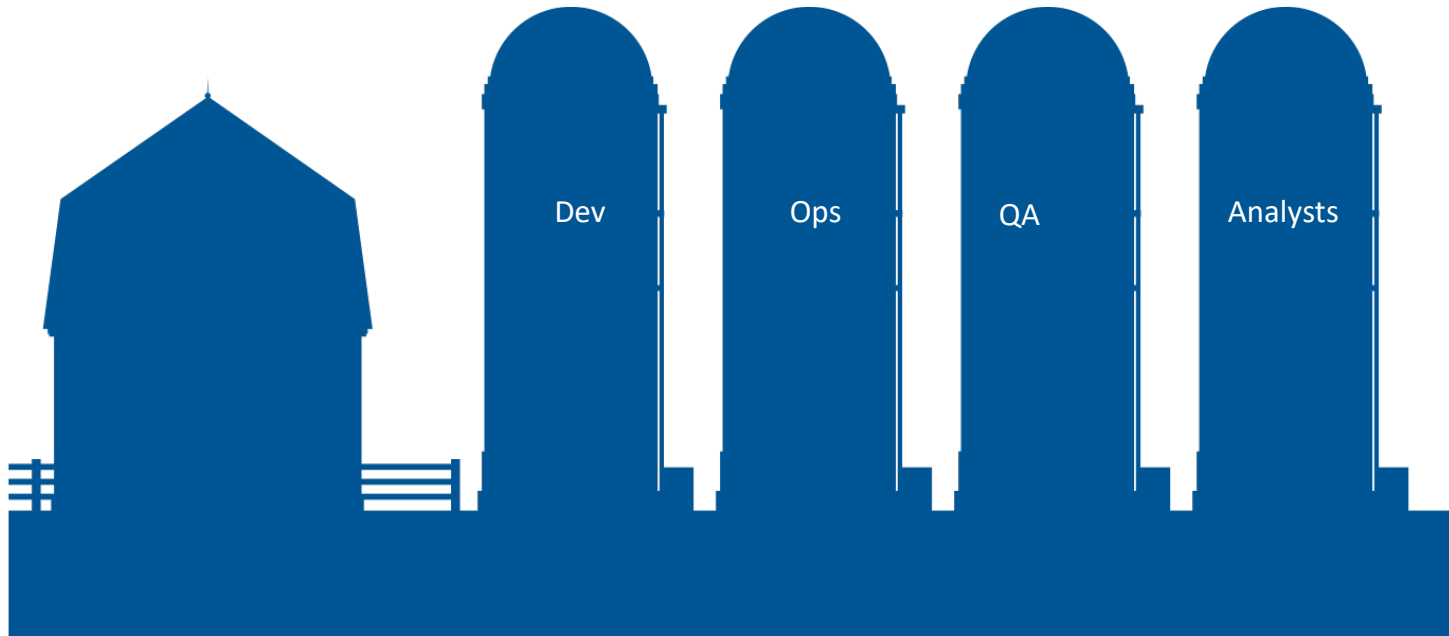


Agile Method



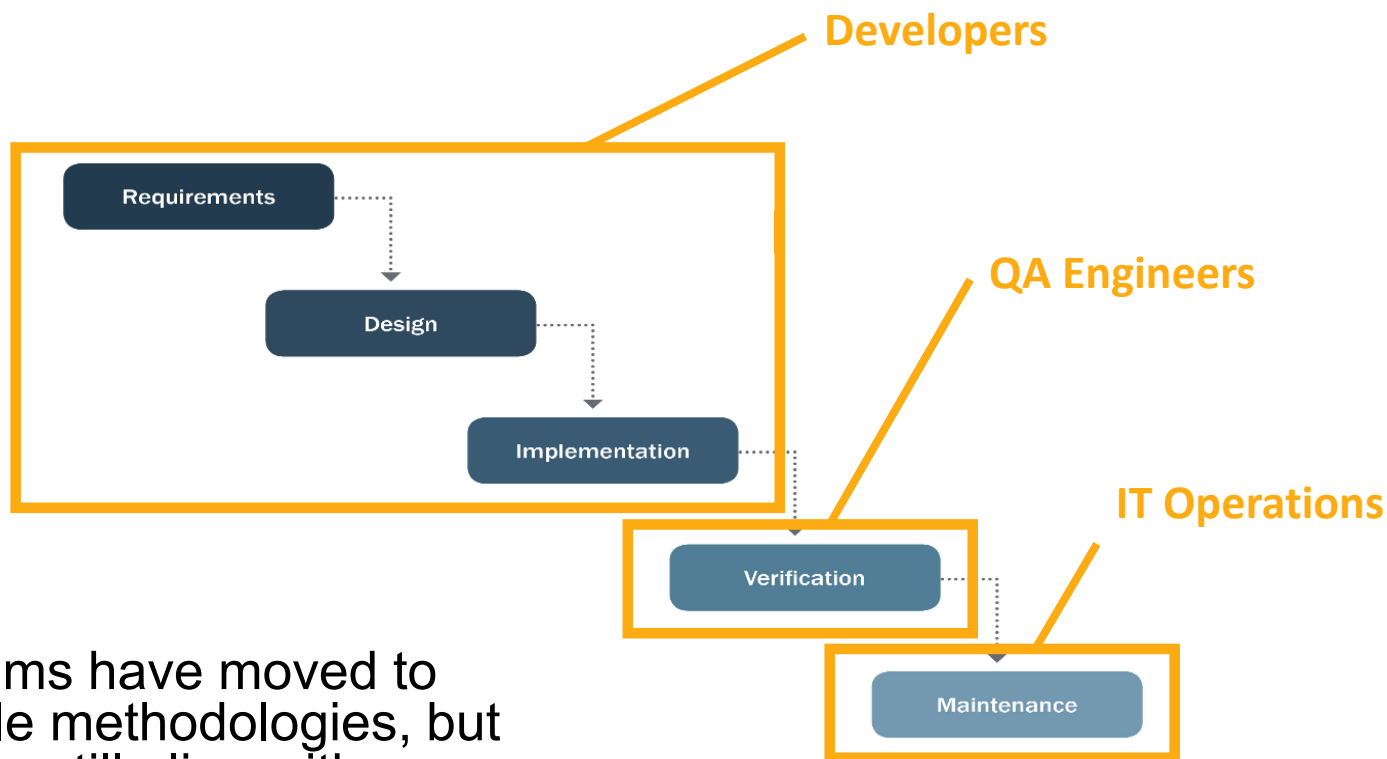


Silos Block Collaboration





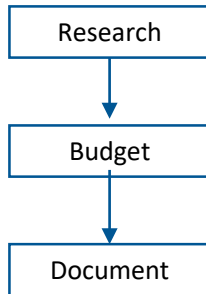
Silos Reinforce Waterfall



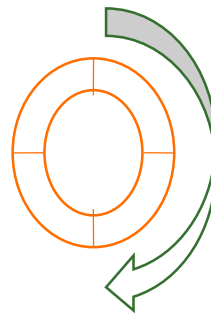
Teams have moved to Agile methodologies, but roles still align with waterfall methods

Water - Scrum - Fall

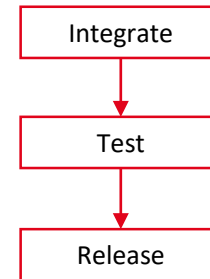
Business



Development



QA
Operations



Jez Humble, https://youtu.be/L1w2_AY82WY
Dave West, <http://sdtimes.com/analyst-watch-water-scrum-fall-is-the-reality-of-agile/>

DevOps is an Extension of Agile Thinking

Agile

Embrace constant change

Embed Customer in team to internalize expertise on requirements and domain

DevOps

Embrace constant testing, delivery

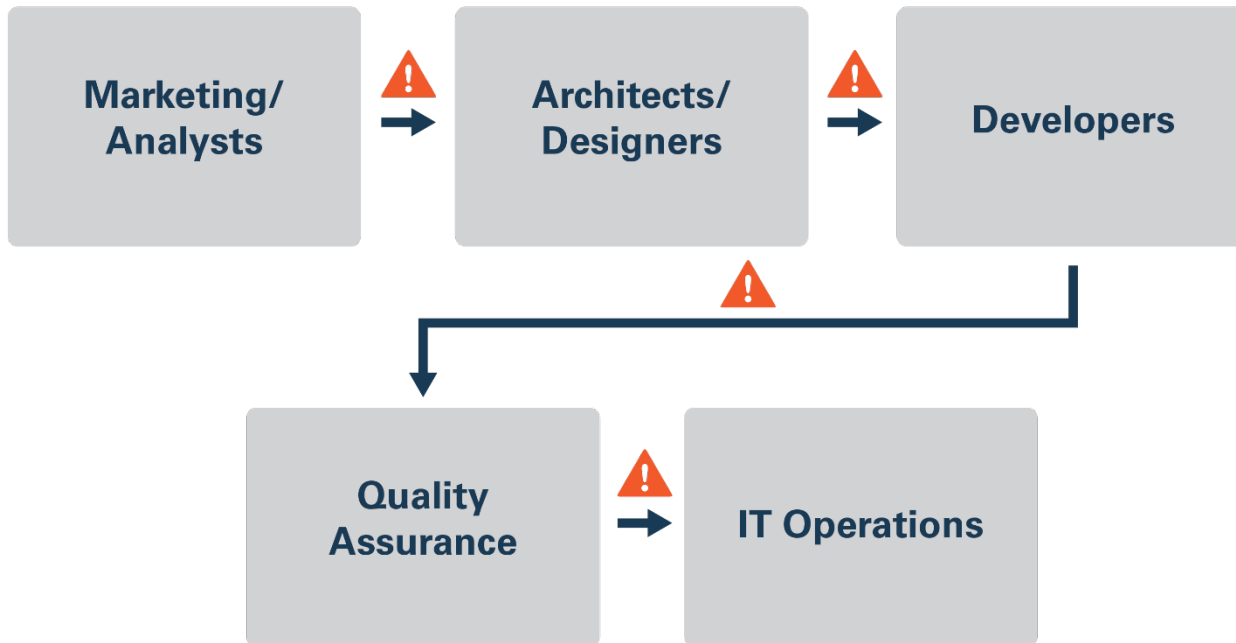
Embed Operations in team to internalize expertise on deployment and maintenance



Polling ?

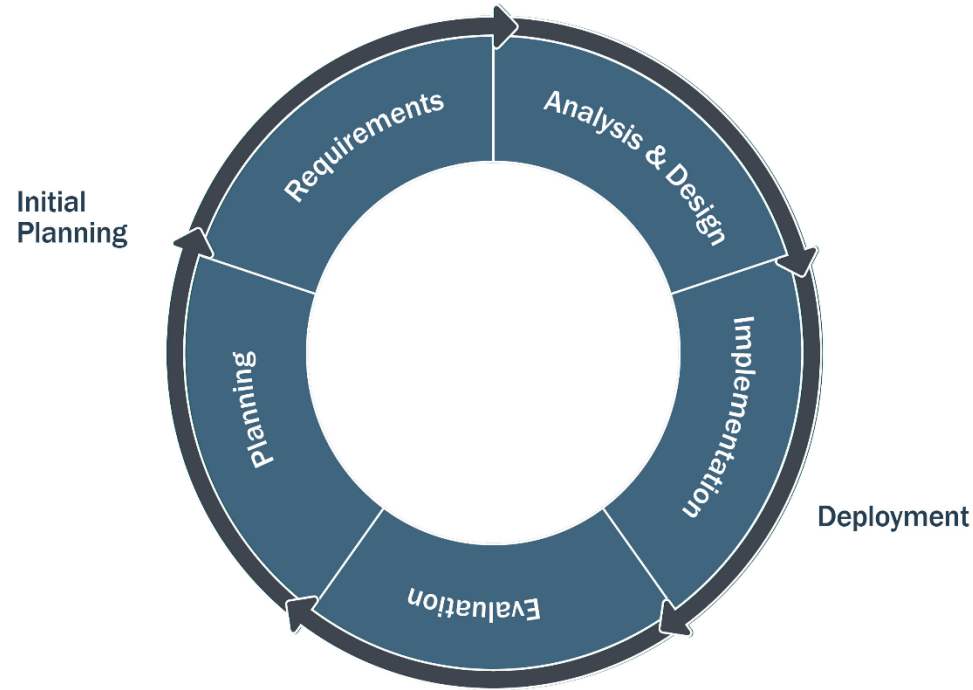
Does your organization follow DevOps process and methodologies?

Every Transition of the System is a Risk



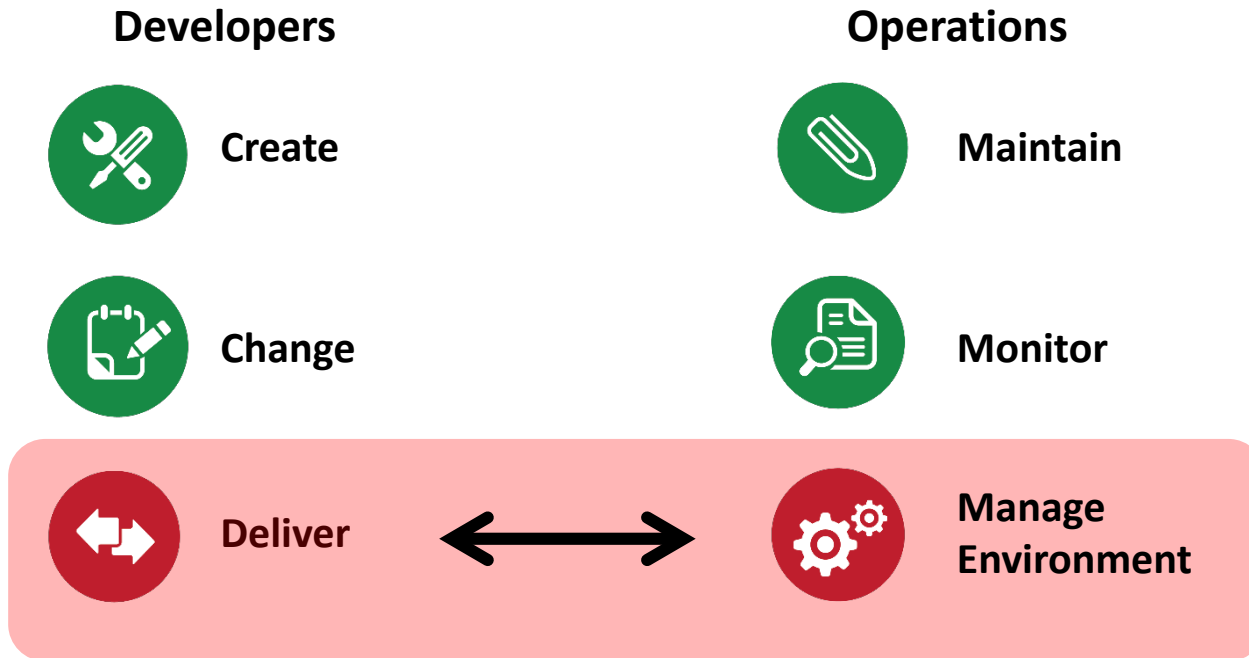
Agile Means Constant Transition

Agile Method





Significant Collaboration Is Needed Where Paths Intersect



To address these pain points, DevOps promotes Collaboration

Heavy collaboration between Dev and Ops on:

- Design / Architecture decisions
- Environment / Network configuration
- Deployment planning
- Code Review

Constantly available open communication channels:

- Dev and Ops together in all project meetings
- Chat/Email/Wiki services available to all team members
- Dev / Ops report together as one project team



An Engaged, Cross-Functional team is needed

Early involvement of experts

- Ops = experts in maintainability and deployability

Complete engagement

- Don't bring Ops Engineers in as consultants – make them first-class team members with same success criteria as devs

Break down organizational silos

- Enable and require constant communication

DevOps Aims to Increase...

...the pace of **innovation**

...**responsiveness** to business needs

...**collaboration**

...software **quality**





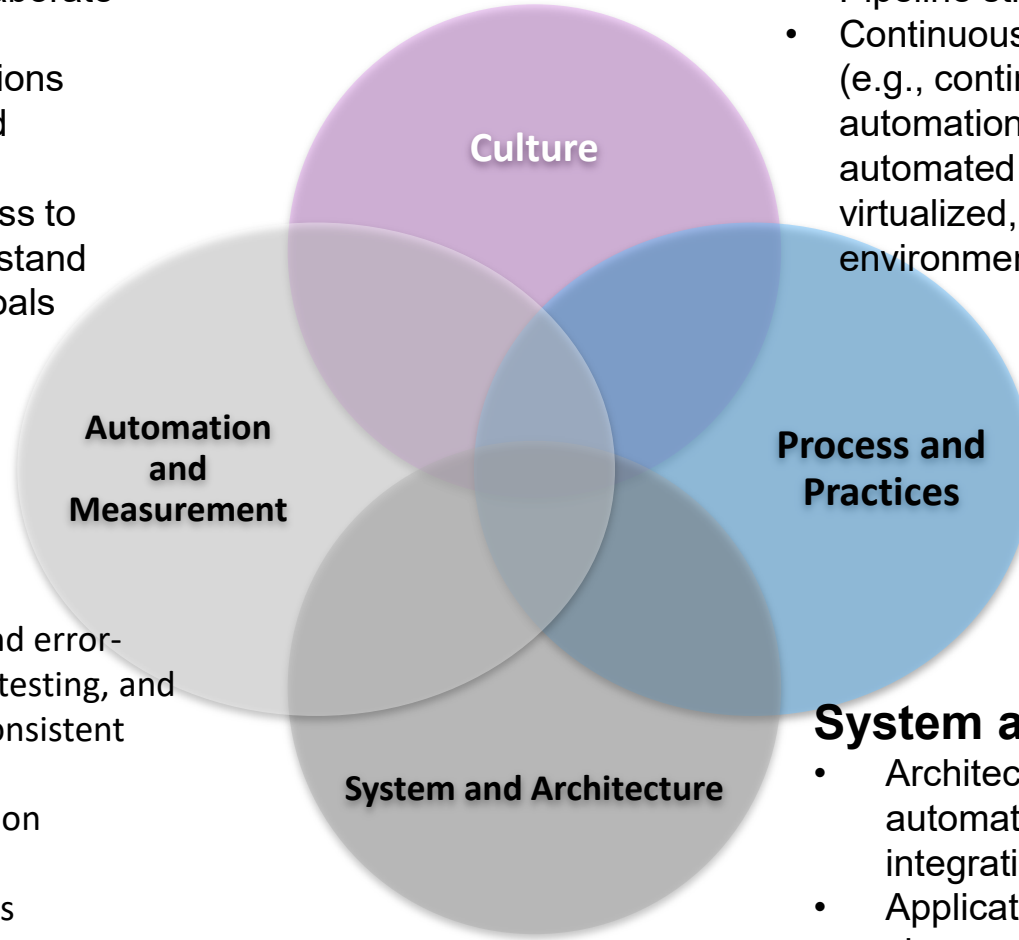
Multiple Dimensions of DevOps

Culture

- Developer and Ops collaborate (Ops includes security)
- *Developers* and Operations support releases beyond deployment
- Dev and Ops have access to stakeholders who understand business and mission goals

Process and Practices

- Pipeline streamlining
- Continuous-delivery practices (e.g., continuous integration; test automation; script-driven, automated deployment; virtualized, self-service environments)



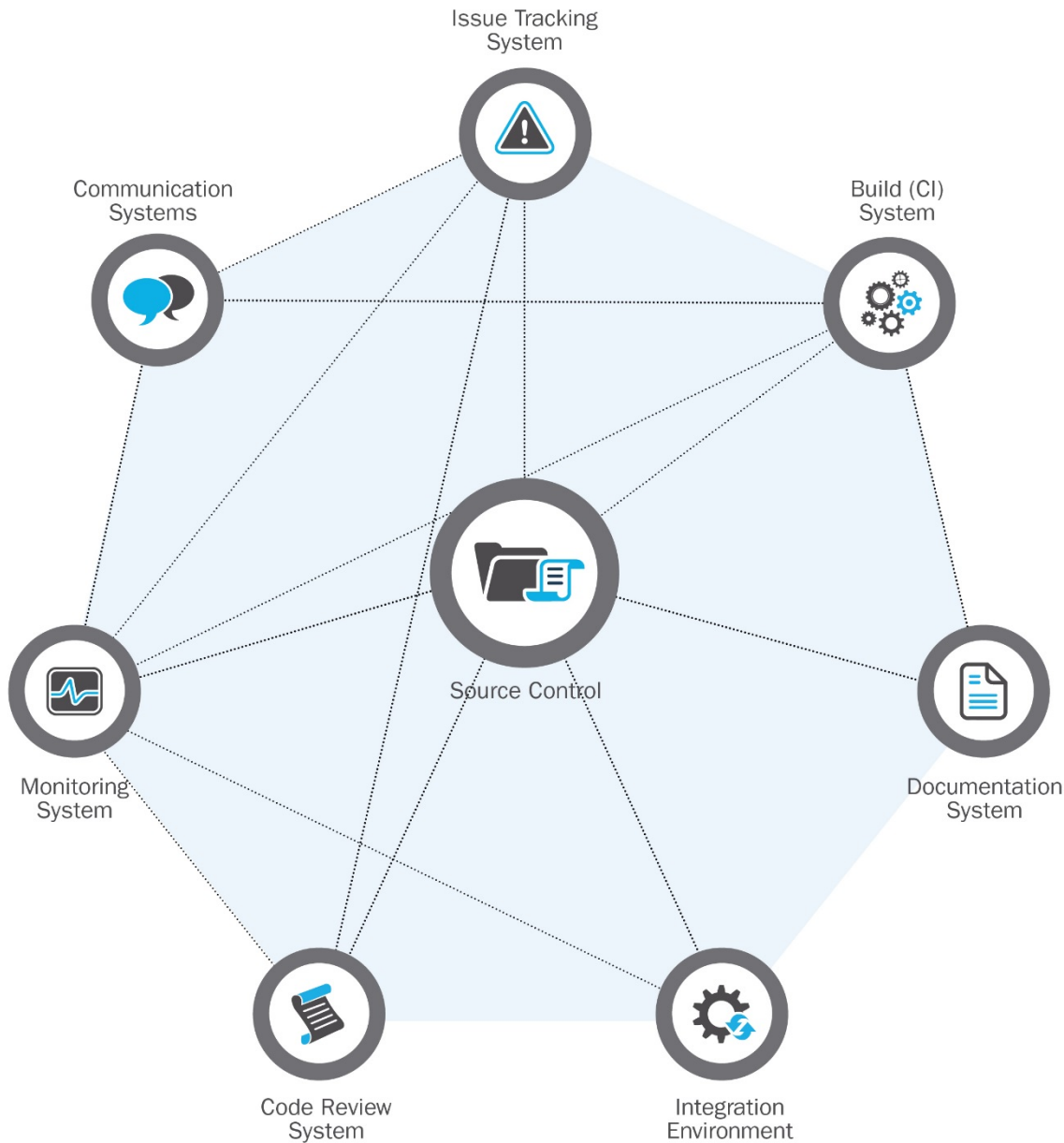
Automation/ Measurement

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

System and Architecture

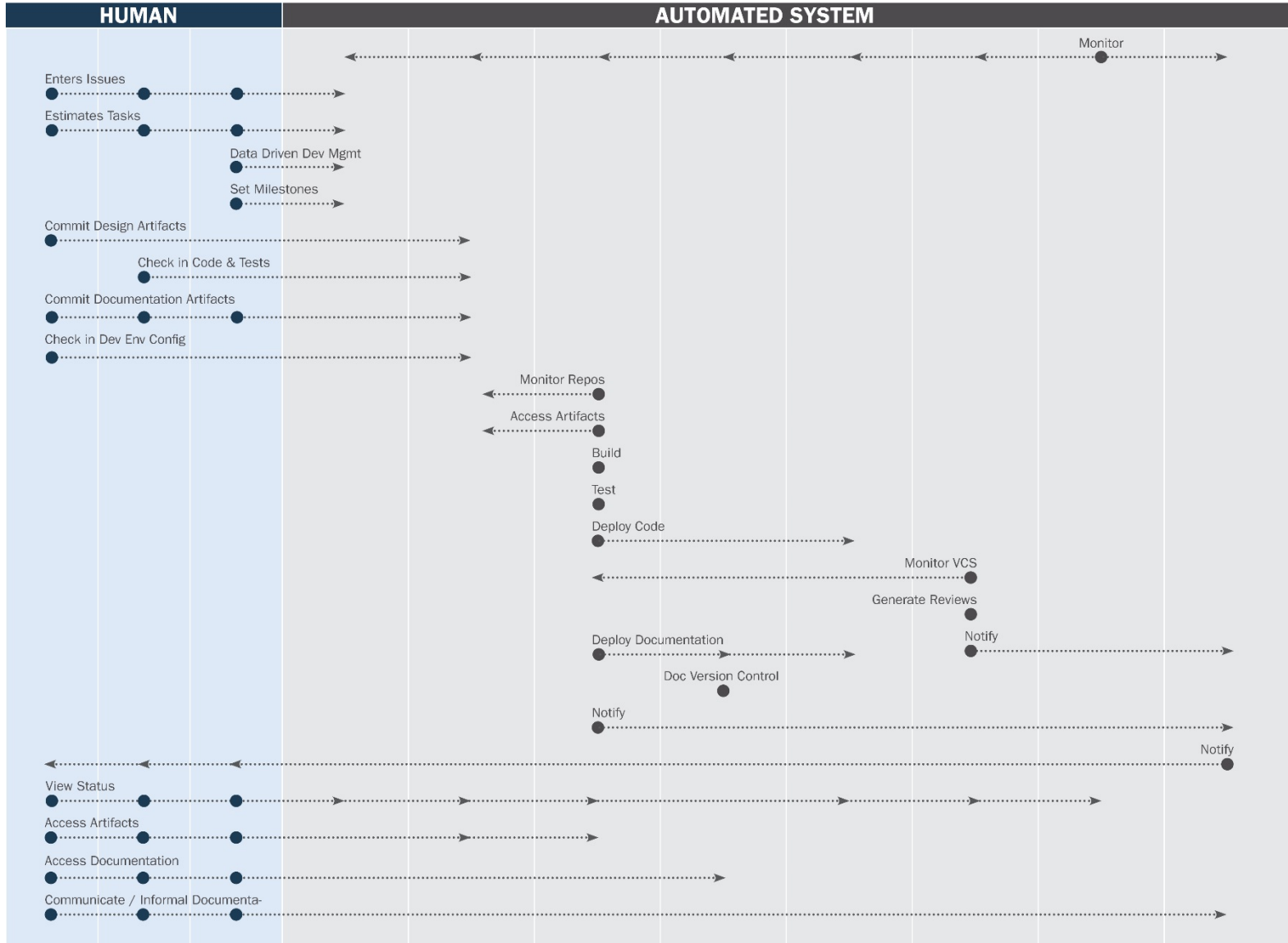
- Architected to support test automation and continuous-integration goals
- Applications that support changes without release (e.g., late binding)
- Scalable, secure, reliable, etc.





Integration and communication, even among tools, is the key to integrate Security into Development Platform!







Building Secure Solutions
DevOps Lesson Learned

Polling ?

Do you have Security Ops Team as part of development activities?

For security teams, the world has changed in three fundamental ways:

- Agility means code deployment is trending to near-instantaneous
- Security is no longer the gatekeeper to deployment
- If security is a blocker, it will be routed around



Near-instantaneous deployment?





A simulation of deploying code in the waterfall model



What is this shifting to?



An agility example: Etsy pushes to production **50 times a day** on average



Constant iteration **in production** via feature flags, ramp ups, A/B testing





But doesn't the
rapid rate of
change mean
things are less
secure?!





Actually, the opposite is true



The key to realize is vulnerabilities occur in
all development methodologies



The key to realize is vulnerabilities occur in
all development methodologies

...But there's no such thing as an out-of-
band patch in continuous deployment



Compared to:

“We’ll rush that security fix. It will go out ... in about 6 weeks.”

- Former vendor at Etsy



Polling ?

Do you believe that the DevOps process, mainly Continuous Delivery is a barrier for application security?

What makes continuous deployment safe?



What makes continuous deployment safe?

Visibility

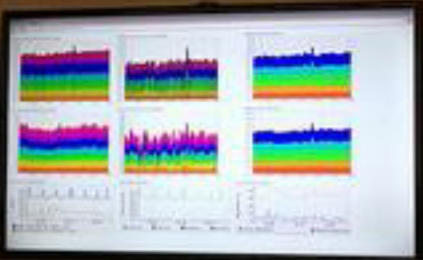


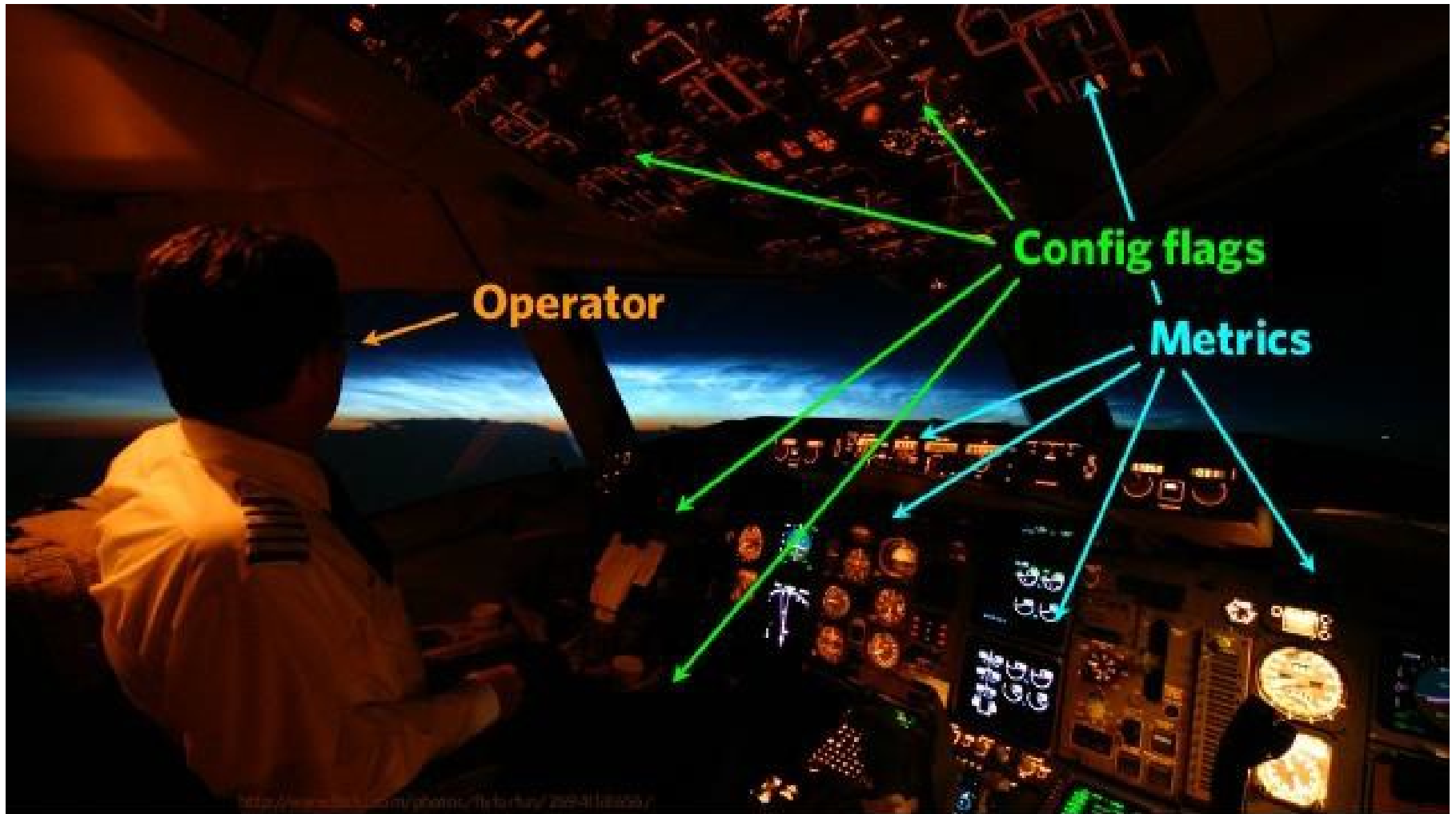


Thank you for esing.



Service	Status	Version	IP	Port	OS
ssh	Up	7.6p1	10.0.2.15	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.16	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.17	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.18	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.19	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.20	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.21	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.22	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.23	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.24	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.25	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.26	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.27	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.28	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.29	22	Ubuntu 14.04 LTS
ssh	Up	7.6p1	10.0.2.30	22	Ubuntu 14.04 LTS





Source: <http://www.slideshare.net/mikebrittain/advanced-topics-in-continuous-deployment>



Software Engineering Institute

Carnegie Mellon University

Security Practitioner Perspective on DevOps for Building Secure Solutions,

October 19th, 016

© 2016 Carnegie Mellon University

The same hard lessons are slowly
shifting to security



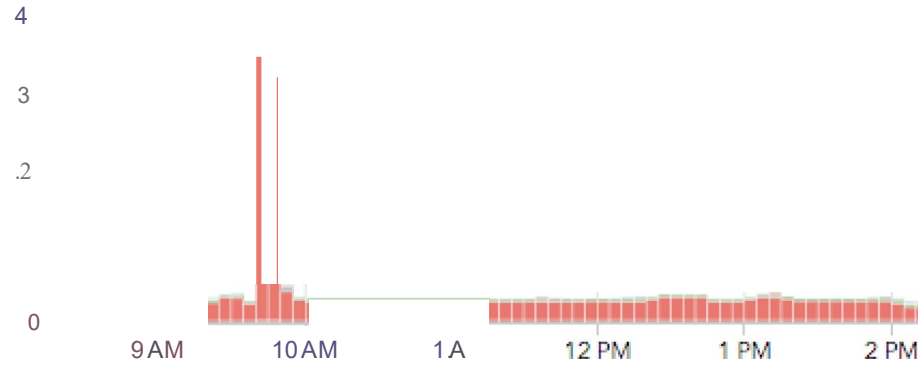
Ex: Which of these is a quicker way to spot an attack?



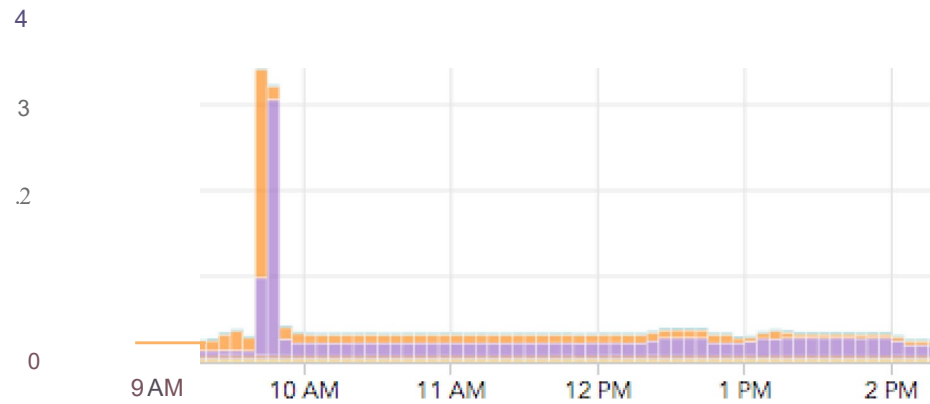
```
se.css" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:10.0) Gecko/20100101 Firefox/10.0" - - - - - [redacted] - - - - - 16951 [redacted]
- - - - [20/Feb/2012:22:32:10 +[redacted]] "GET /images/sprites/buttons-master.png HTTP/1.1" 304 - "http://[redacted]assets/dist/88166671/css/modules/buttons-new.css" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:10.0) Gecko/20100101 Firefox/10.0" - - - - - [redacted] - - - - -
- 12156 [redacted]
- - - - [20/Feb/2012:22:32:10 [redacted]] "GET /images/sprites/sprinner16.gif HTTP/1.1" 304 - "http://[redacted]t/ossets/dist/88166671/css/base.css" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:10.0) Gecko/20100101 Firefox/10.0" - - - - - [redacted] - - - - - 18810 [redacted]
- - - - [20/Feb/2012:22:32:10 [redacted]] "GET /assets/dist/88166671/js/convo/threads.js HTTP/1.1" 200 61743 "http://[redacted]/conversations?ref=si_con" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:10.0) Gecko/20100101 Firefox/10.0" - - - - - [redacted] - - - - - 834687 [redacted]
- - - - [20/Feb/2012:22:32:10 [redacted]] "GET /assets/dist/88166671/js/bootstrap/common.js HTTP/1.1" 200 127238 "http://[redacted]/conversations?ref=si_con" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:10.0) Gecko/20100101 Firefox/10.0" - - - - - [redacted] - - - - - 928201 [redacted]
- - - - [20/Feb/2012:22:32:11 [redacted]] "GET /ossets/dist/88166671/js/overlays/external-link.js HTTP/1.1" 200 487 "http://[redacted]/conversations?ref=si_con" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:10.0) Gecko/201
```



Attacks >



Anomalies >



Increase agility by surfacing security
visibility for **everyone**, not just the
security team



Having to talk to security to get
security awareness causes
delays



Having to talk to security to get security awareness causes delays

Delays get routed around



To embrace agility, security has to
decentralize



Lessons Learned:

- Embracing DevOps/Agile/Continuous Deployment helps not harms security
- Visibility is the key to moving quickly and safely
- You (in the general case) are never going to be able to hire enough staff, so steal everyone else's

More on SEI DevOps Blog

<https://insights.sei.cmu.edu/devops>

<https://signalsciences.com/resources/>

Thank you!



zane@signalsciences.com

hyasar@sei.cmu.edu

@zanelackey

@securelifecycle

