

# RSA® Conference 2016

Singapore | 20-22 July | Marina Bay Sands



Connect **to**  
Protect

SESSION ID: SDS1-F03

## The Need for Speed: AppSec in a DevOps World

**John B. Dickson, CISSP**

Principal  
Denim Group, Ltd.  
@johnbdickson



#RSAC



- DevOps Defined
  - What's Driving DevOps?
- The evolution of application development and application security
- Case Studies: Etsy and Netflix
- How Application Security Remains Relevant in a DevOps World



# Applying What You Will Learn Today



#RSAC

- Next week you should:
  - Be immediately comfortable having a discussion about DevOps and application security with your colleagues and management
- In the first three months following this presentation you should:
  - Understand your organization's DevOps strategy
  - Apply initial application security strategies to your organizations DevOps practices
- Within six months you should:
  - Be a partner with your business units to rapidly develop software while addressing security risks throughout the process



# John's Background



- Application Security Enthusiast
- Helps CSO's and CISO's with Application Security Programs
- ISSA Distinguished Fellow
- Security Author and Speaker
- 20 years Experience Across Multinational Corporations





- Professional services firm that works closely with companies on matters of software risk
  - Web, mobile, and cloud application assessments
  - Application vulnerability mitigation
  - Classroom secure developer training
- Network & information security services
- Outsourced managed security services
- Developed [ThreadFix](#) – application vulnerability platform



**ThreadFix**

Powered by Denim Group





- DevOps is a practice that:
  - Emphasizes the tight collaboration and communication of both software developers and IT operations staff
  - Focuses on automating the process of software delivery and infrastructure changes
  - Aims at establishing a culture and environment where building, testing, and releasing software, can happen rapidly, frequently, and more reliably



# Aspects of DevOps



#RSAC

- Focuses on time to market over virtually every other requirement
- Focuses on continuous improvement
- Software quality and auditability valued – but as a by-product of speed





- **Continuous Integration (CI)** is a development practice that requires developers to **integrate** code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.
- **Continuous Delivery** is the natural extension of Continuous Integration: an approach in which teams ensure that every change to the system is releasable, and that we can release any version at the push of a button. Continuous Delivery aims to make releases boring, so we can deliver frequently and get fast feedback on what users care about
  - Source: “Continuous Integration: Improving Software Quality and Reducing Risk,” Paul Duvall



# Potential Components of a Secure CI/CD



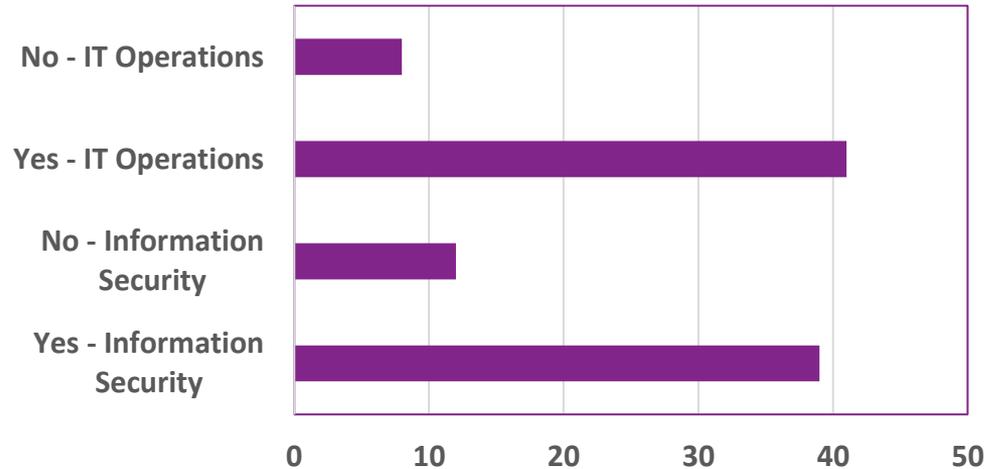
- Code repository (Git, Subversion)
- CI/CD server (Jenkins, Bamboo)
- Build server(s)
- Unit test suite (JUnit)
- Functional test suite (Selenium)
- Defect tracker
- Application Vulnerability Management Platform

# What is Driving DevOps?



- Time-to-Market advantages
- Demand of higher quality software products
- Cost concerns
- Key thought: Like cloud, DevOps will come from business units responding to competitive pressures, not IT or outside pressure

- Do you believe your information security policies/teams are slowing IT down?

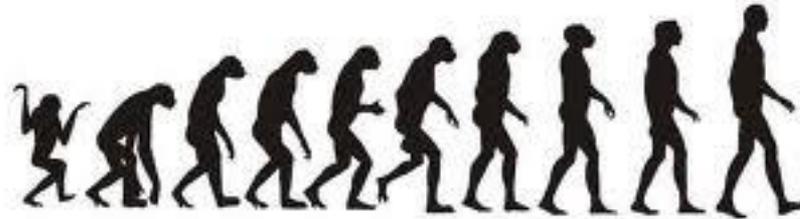


- Source: Gartner: Integrating Security in Devops: DevSecOps, Neil McDonald, 2016

# The Evolution of Application Development



#RSAC



Where are we in the evolution of software development?



- Waterfall
- Agile Software Development Methodology
  - Scrum
  - Extreme Programming (XP)
- Just to Name a few...



- Waterfall
  - Linear with distinct goals in each phase of development
  - Requirements laid out up front by business units
  - Clear separation between business units and software development team
  - Deployments typically infrequent and involve close coordination with development and operations teams
  - Criticized as being too inflexible and not taking in account change within a project



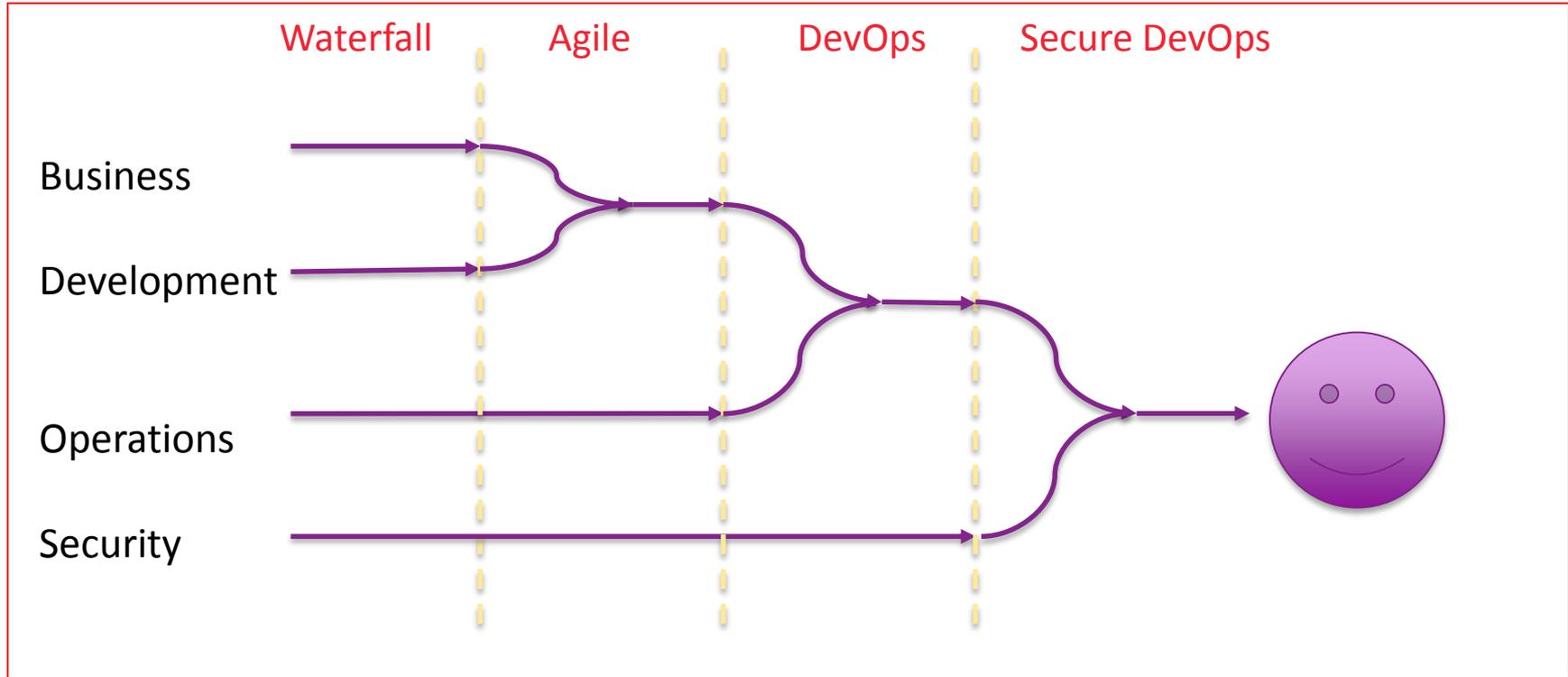
- Agile
  - Iterative software development in short “sprints” of 1-4 weeks
  - Focus to produce working software that allows business teams to provide better feedback after each sprint
  - Business teams conduct tradeoff analysis and adapt requirements after each sprint (and are willing to give up requirements)
  - More frequent interaction between software development, test, and business teams



# How Did We Get to DevOps?



#RSAC



# The State of Application Security



#RSAC

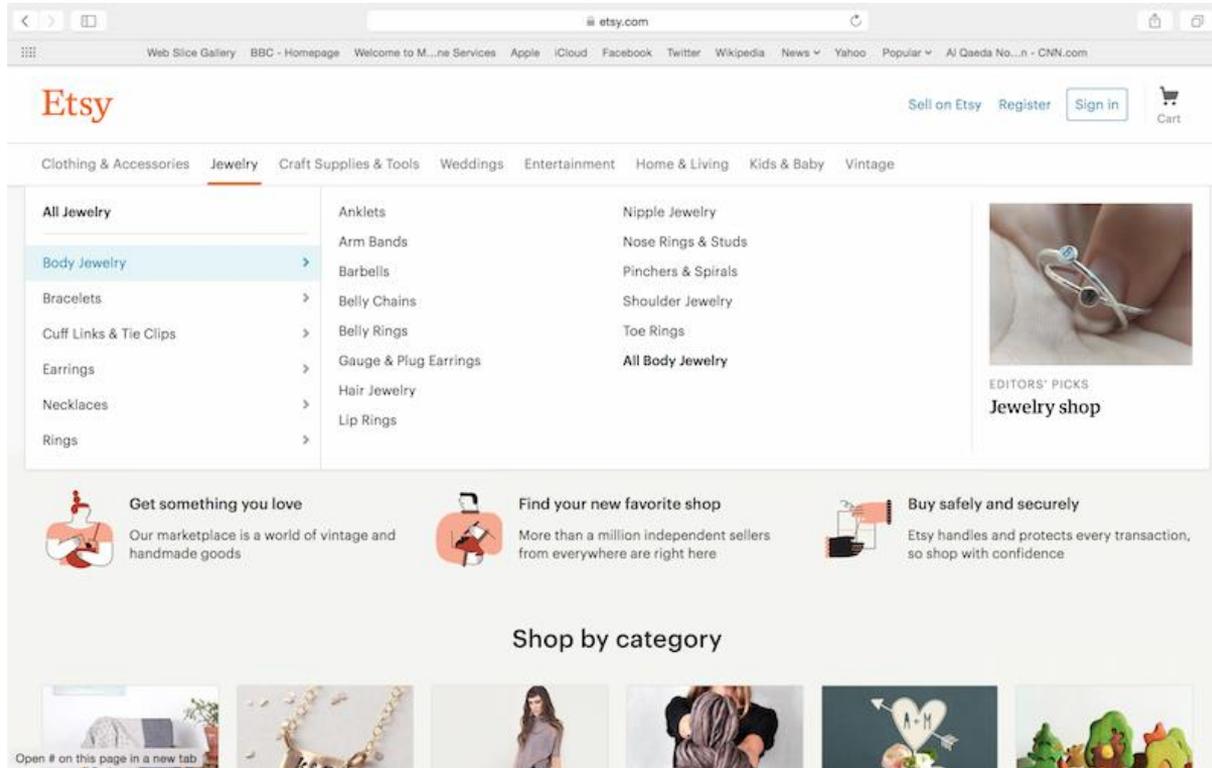
- Organizations have become better at identify web application vulnerabilities via automated scanning
- Automation still only catches 30-50% of application vulnerabilities
- Organizations have become better at identifying application vulnerabilities than fixing them
- Much of the effort involves testing and SDLC improvement
- Chasm still exists between security and development teams



# Case Study: Etsy



#RSAC



# Case Study: Etsy



#RSAC





- Etsy pushes to production 30 times a day on average
- Schema changes weekly
- Code reviews before commits
- Automated tests before deploy
- Verification conducted frequently and in small batches
- No release managers
  - Source: "Continuous Delivery: The Dirty Details," Mike Brittain, Etsy



# Case Study: Etsy – Key Takeaways

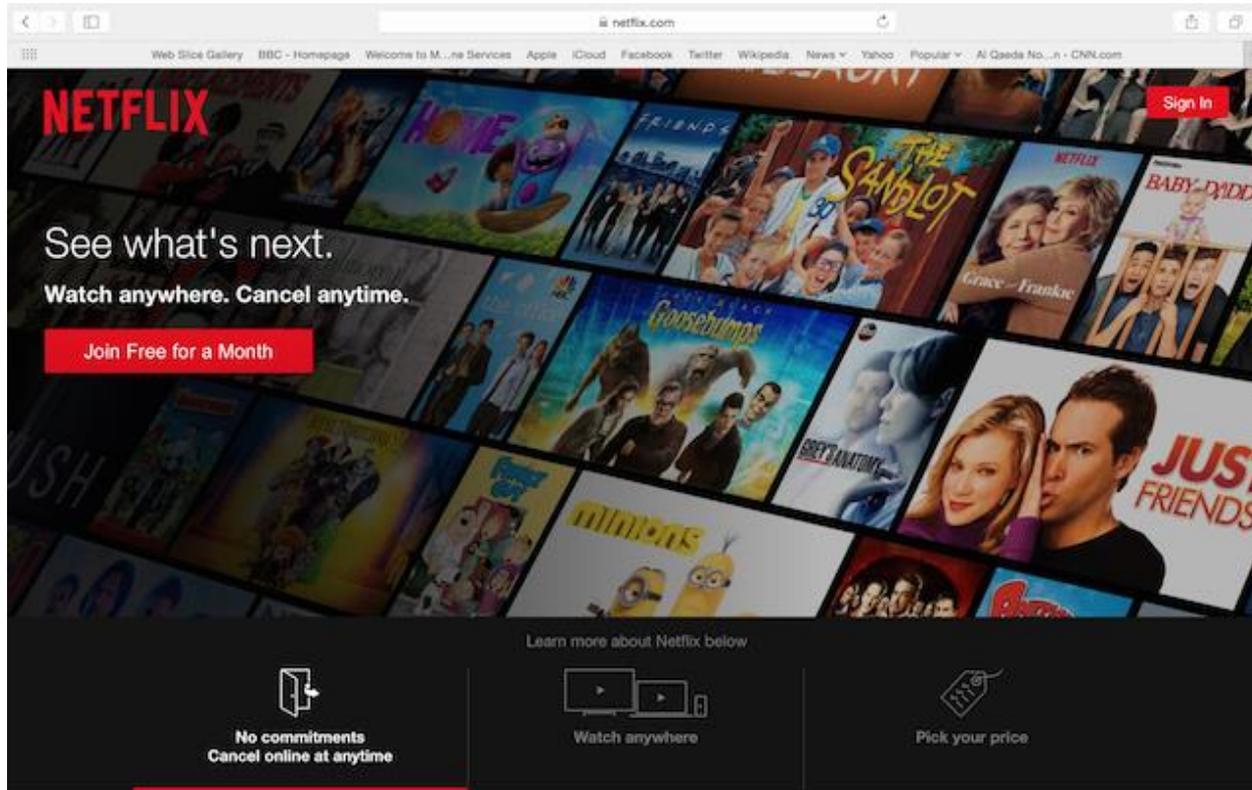


- Make things safe by default
  - Detect risky functionality / Focus on efforts
  - Automate as much as you can
  - Know when the house is burning down
- Source: “Effective Approaches to Web Application Security, Zane Lackey, <http://www.slideshare.net/zanelackey/effective-approaches-to-web-application-security>

# Case Study: Netflix



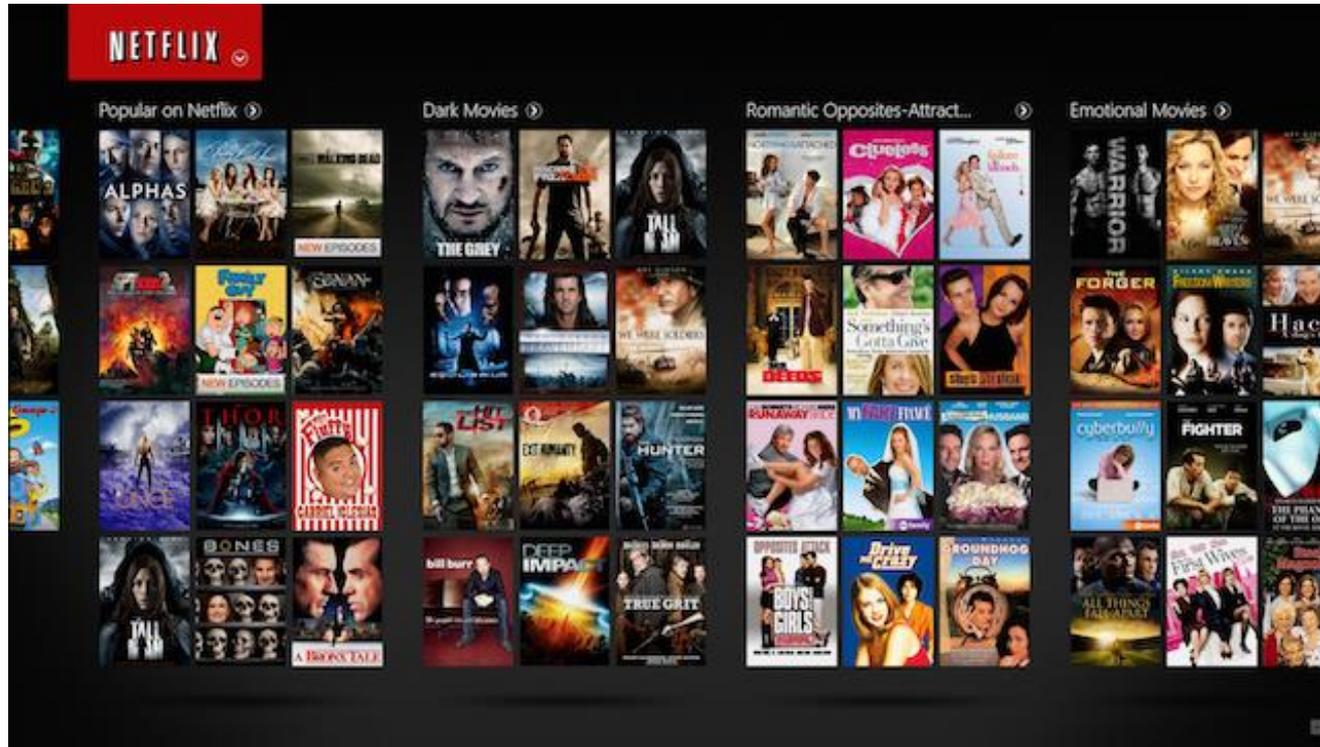
#RSAC



# Case Study: Netflix



#RSAC



# Case Study: Netflix



#RSAC

- Everything is built for “three”
- Fully automated build tools to test and make packages
- Fully automated machine image bakery
- Fully automated automated image deployment
- Independent teams responsible for both Dev and Ops



# Case Study: Netflix



#RSAC

- All systems choices assume some part will fail at some point
- Availability over consistency
- Scanning for vulnerabilities in production via the “Simian Army”



# How Application Security Remains Relevant in a DevOps World



#RSAC

- Pulling a Tiger by the Tail?



# How Application Security Remains Relevant in a DevOps World



- Understand that you will miss things
- Software will be deployed without your knowledge and not security tested (always)
- You will have functionality in your production environment you don't understand
- Understand your job just got harder
- And you can't say "no!"

# Understand There are Competing World Views of DevOps and Security



#RSAC

- Do you try to adapt current application security/SDLC approaches with more automation?
- OR
- Do you accept that you can only be prepared to improvise when code is in production



# Where do You Go from Here?



# DevOps Concepts if You Take Adaptation Approach



#RSAC

- Automate every security process possible
  - Squeeze application testing cycles and automate entire process
  - Fully automate application vulnerability resolution process
- Consider new technologies such as IAST/RASP
- Incrementally increase application monitoring in production environments – standardize & automate





- Focus on testing in production environments
  - Create processes and scanning systems to tear down vulnerable functionality
  - Recognize that production is where you might first learn of new features!
- Recognize application attack patterns in production environments via big data
  - Fix vulnerability!





# 2016 Gartner Recommendations

- If you haven't already, get involved in DevOps initiatives
- Remain true to DevOps philosophy: Teamwork and transparency
- Require security and management vendors to:
  - Fully API-enable their platform services
  - Provide out-of-the box support for common DevOps toolchain environments
  - Provide out-of-the box support for containers and management systems
- Make OSS software module identification and vulnerability scanning a priority in 2016
- Don't use containers spanning trust levels on same system



# Apply What You Learned Today



#RSAC

- Next week you should:
  - Be immediately comfortable having a discussion about DevOps and application security with your colleagues and management
- In the first three months following this presentation you should:
  - Understand your organization's DevOps strategy
  - Apply initial application security strategies to your organizations DevOps practices
- Within six months you should:
  - Be a partner with your business units to rapidly develop software while addressing security risks throughout the process



- “Continuous Integration: Improving Software Quality and Reducing Risk,” Paul Duvall
- “Continuous Delivery: The Dirty Details,” Mike Brittain, Etsy
- “DevOpsSec: Delivering Secure Software Through Continuous Delivery,” Jim Bird
- “Effective Approaches to Web Application Security, Zane Lackey, Signal Science
- “Integrating Security in DevOps: DevSecOps,” Neil McDonald, Gartner

## Questions and Answers

**John B. Dickson**

**@johnbdickson**

