Das „s“ in DevOps steht für Security

Fallstudie: Sicherheit in agiler Softwareentwicklung

Jan Harrie <jharrie@ernw.de>

#whoami - Jan

- Security Consultant @ERNW GmbH
- Former Security Analyst/Pentester/WebApp-Monkey
- M.Sc. IT-Security TU Darmstadt

- Interests:
  - Container, DevOps & Orchestration Solutions
  - Gardening

ERNW
providing security.

Agenda

- Motivation
- Initial Situation
- State of the Issue
- Security in Agile SW Development
- Conclusion

Motivation

Integrate security into modern development lifecycles and make security suitable, accessible, and measurable for each project

In other words …

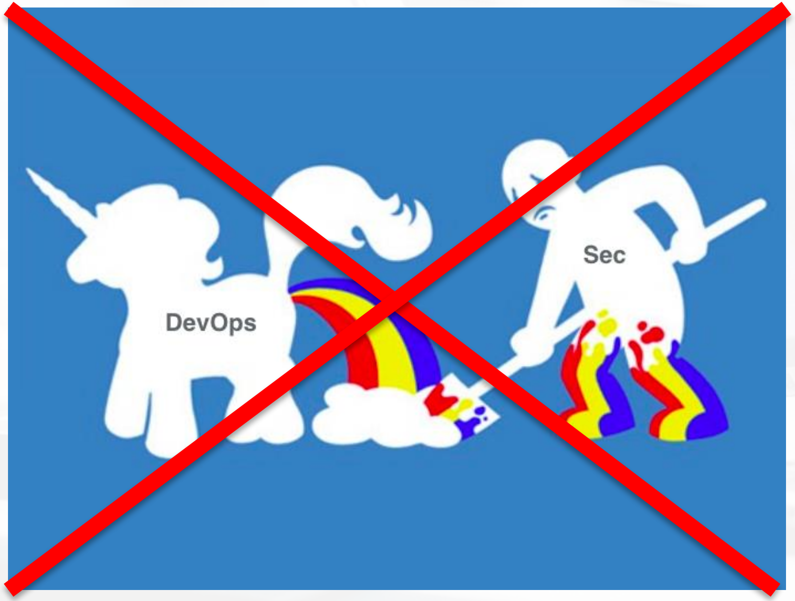


Image source: https://twitter.com/petecheslock/status/595617204273618944t

Initial Situation

Traditional SW development approach, no further specified security considerations

- Missing guidance
- Missing technical support
- Limited requirements
- Limited defaults

Past: Waterfall Model

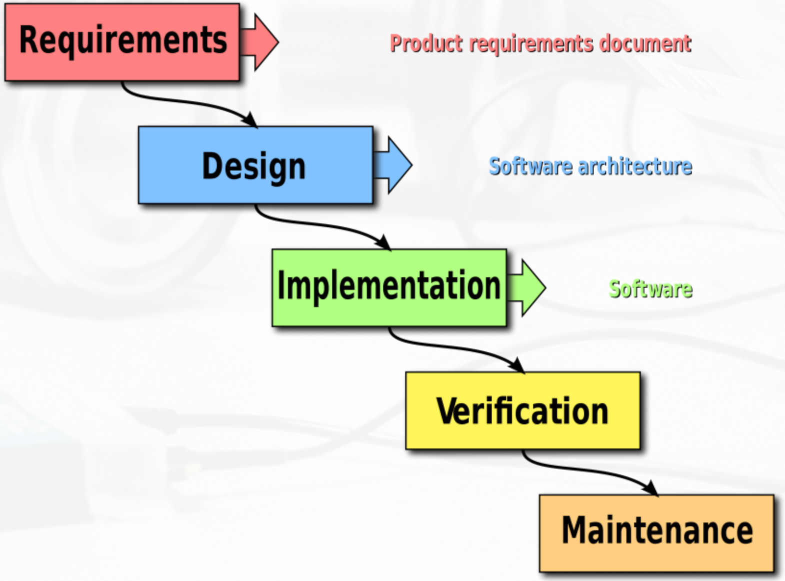
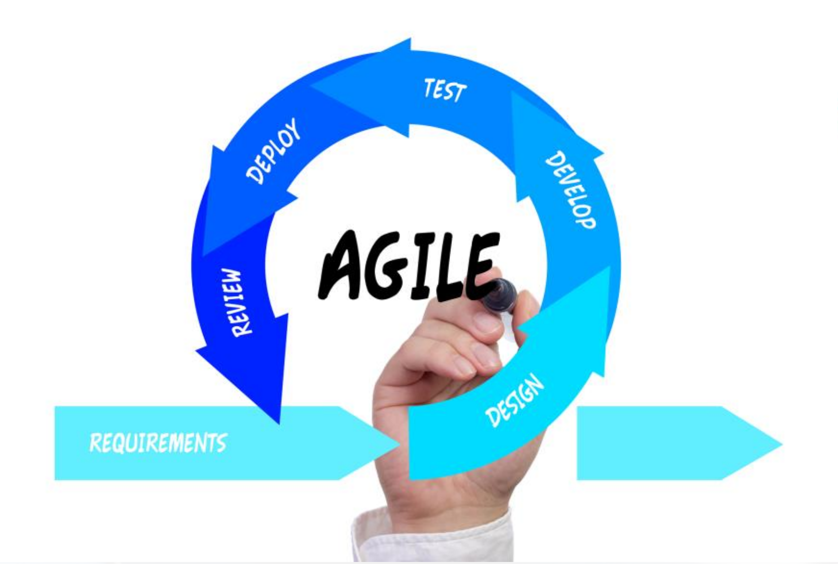


Diagram source: https://en.wikipedia.org/wiki/Waterfall_model#/media/File:Waterfall_model.svg
Image source: https://alln-extcloud-storage.cisco.com/ciscoblogs/5ad679887cc5d.jpg

Now: Agile Software Development



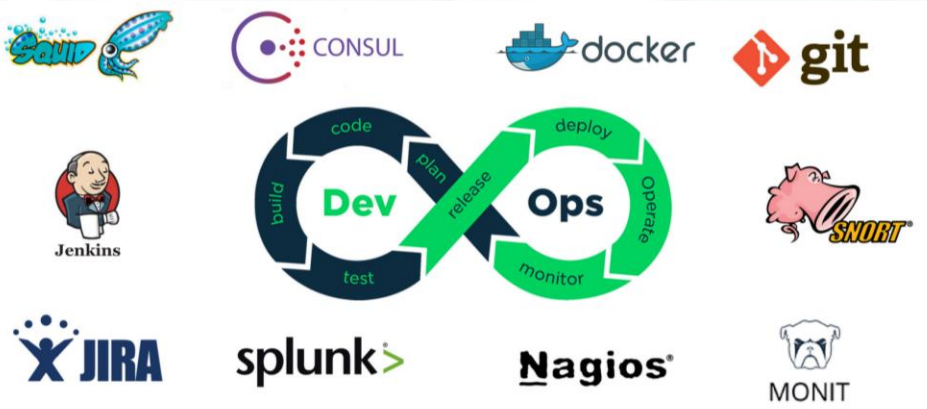
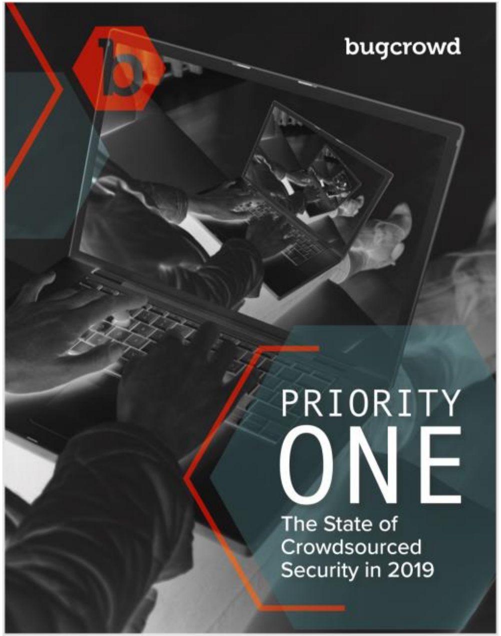


Diagram source: https://www.proficientz.com/wp-content/uploads/2018/08/agile-software-development.jpg
Image source: https://www.tektutes.com/wp-content/uploads/2018/08/Top-Devops-Tools.png



bugcrowd

PRIORITY ONE

The State of Crowdsourced Security in 2019

hackerone

THE HACKER POWERED SECURITY REPORT 2019

The study on the hacker-powered security ecosystem

@try_to_hack: first hacker to achieve $1M in bounties

State of the Issue

... a look into the threat landscape

OWASP TOP10 2013

initial proposal

- A1 Injection
- A2 Broken Authentication and Session Management
- A3 Cross-Site Scripting (XSS)
- A4 Insecure Direct Object References (IDOR)
- A5 Security Misconfiguration
- A6 Sensitive Data Exposure
- A7 Missing Function Level Access Control
- A8 Cross-Site Request Forgery (CSRF)
- A9 Using Components with Known Vulnerabilities
- A10 Unvalidated Redirects and Forwards

[9] OWASP TOP10 2013

OWASP TOP10 2017

re-checked

- A1 Injection
- A2 Broken Authentication
- A3 Sensitive Data Exposure
- *A4 XML External Entities*
- A5 Broken Access Control
- A6 Security Misconfiguration
- A7 Cross-Site Scripting (XSS)
- *A8 Insecure Deserialization*
- A9 Using Components with Known Vulnerabilities
- **A10 Insufficient Logging & Monitoring**

[3] OWASP TOP10 2017

ERNW
providing security.

hackerone

THE HACKER POWERED SECURITY REPORT 2019

HACKER-POWERED SECURITY REPORT 2019

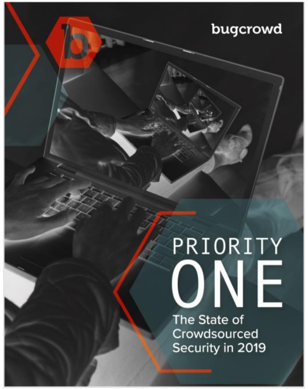
The top 15 vulnerability types platform-wide

- Cross-Site Scripting (XSS)
- Information Disclosure
- Improper Access Control
- Violation of secure Design Principle
- Improper Authentication
- Cross-Site Request Forgery (CSRF)
- Open Redirect
- Business Logic Errors
- Privilege Escalation
- Insecure Direct Object Reference (IDOR)
- Server-Side Request Forgery (SSRF)
- Code Injection
- SQL Injection
- Denial of Service
- Cryptographic

[1] HACKER-POWERED SECURITY REPORT 2019

THE STATE OF CROWDSOURCED SECURITY IN 2019

Top submitted vulnerabilities on web applications



bugcrowd

PRIORITY ONE

The State of Crowdsourced Security in 2019

- Cross-Site Scripting (XSS)
- Information Disclosure
- Improper Access Control
- Violation of secure Design Principle
- Improper Authentication
- Cross-Site Request Forgery (CSRF)
- Open Redirect
- Business Logic Errors
- Privilege Escalation
- Insecure Direct Object Reference (IDOR)
- Server-Side Request Forgery (SSRF)
- Code Injection
- SQL Injection
- Denial of Service
- Cryptographic

[2] THE STATE OF CROWDSOURCED SECURITY IN 2019

Security in Agile SW Development

Steady Quality Improvement

PDCA as overall quality improvement approach – applicable to both, security and agile SW development

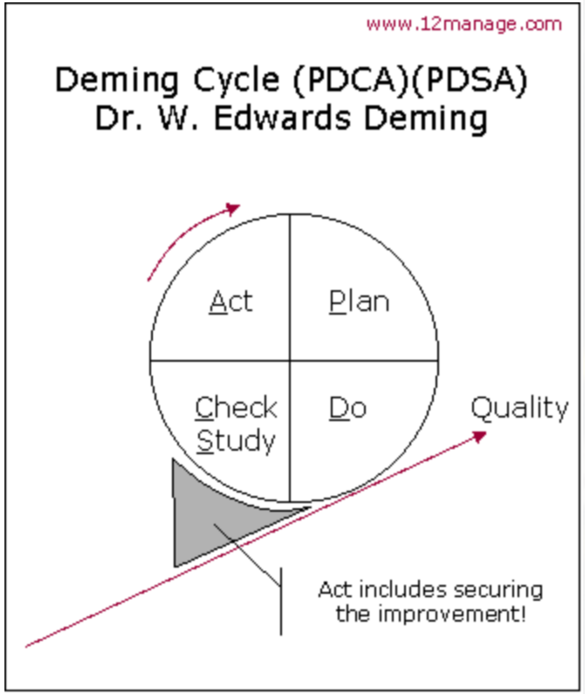

www.12manage.com
Deming Cycle (PDCA)(PDSA)
Dr. W. Edwards Deming
Act
Plan
Check
Study
Do
Quality
Act includes securing the improvement!

Image source: https://www.12manage.com/methods_demingcycle_de.html

Thoughts and Sources

- Industry’s Best Practices
- Agile Manifesto
- DevSecOps Maturity Model
- Standards (ISO27000 et. al.)
- Microsoft SDL
- Open Source Security Testing Methodology Manual (OSSTMM)

ERNW
providing security.

Solution

Central tracking that includes:
- Info, Responsibilities & Deadline
- Risk Assessment
- Status Tracking

Secure Defaults and Templating

Implementation Support

Info, Responsibilities & Deadlines

- Basic Application Information
- Project Roles
- Emergency Contacts
- Remediation Plan
- Application Owner Tasks
- Backup Strategy

Risk Assessment

- Question Categories:
  - Accessibility
  - User Group
  - Authentication
  - Information Criticality
  - Application Complexity
  - Business Criticality
- Base-Score Calculation
- Risk Rating Derivation

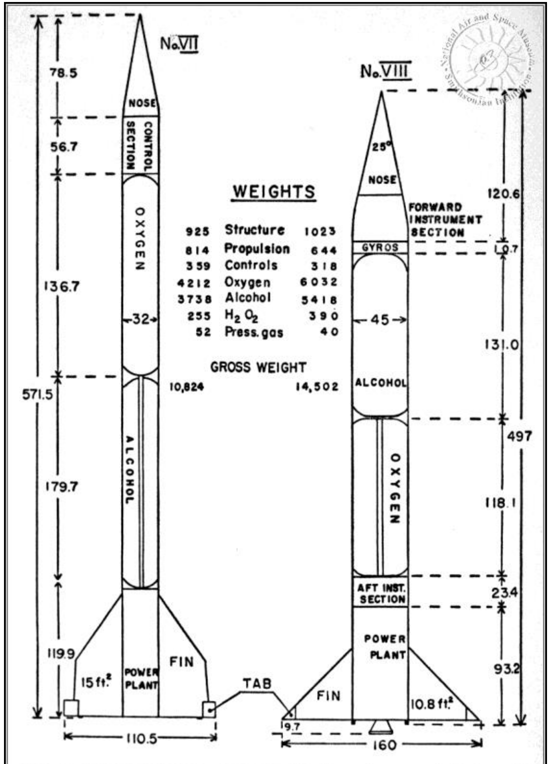


Fig. 47. Viking rockets VII and VIII. All dimensions are in inches; the weights for VIII are those planned for take-off.

Image source: https://commons.wikimedia.org/wiki/File:Viking_rocket_scheme.jpg

ERNW
providing security.

Risk Aligned Security Guidance

Guidance Categories

- Requirements
- Controls
- Design Decisions

Document

- Keep track of decisions made and reasoning
- Opportunity to re-assess decisions and track corresponding evidences
- Visibility of progress

Image source: https://cdn.pixabay.com/photo/2018/10/04/07/41/form-3723114_960_720.jpg

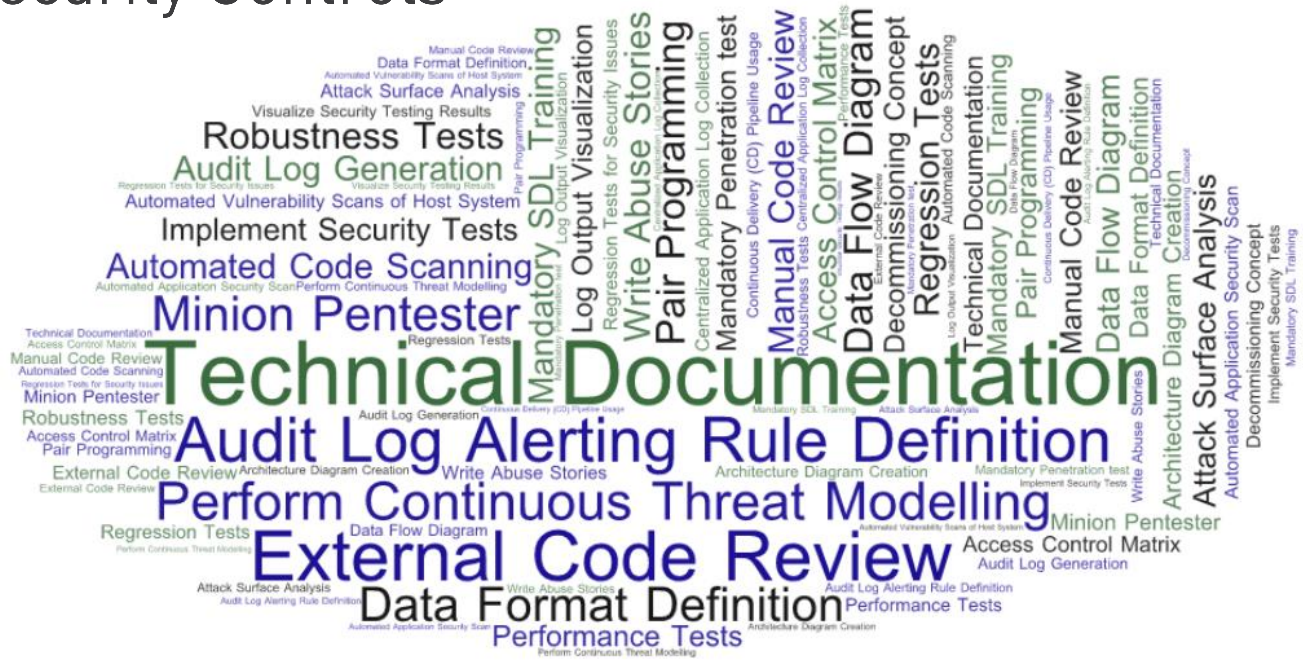
Security Requirements

State the hard facts, e.g.:

- Passwords are individual salted and hashed before storage
- HTTPS communication is always enforced
- Input validation is performed on server-side
- etc.

Re-check justification for not implemented requirements

Security Controls



Security Control Categories

Low

Medium

High

Security Controls: Low

Low | Medium | High

Automatable

- Central Code Repositories
- Automated Builds
- Unified Deployments
- ➡ Secure Scaffolding
- CI Pipeline
- Centralized Infrastructure
- Scans for External Libs
- Hardened Base Images
- Ticket System Integration
- Code Scanning
- Central Application Log Collection
- ➡ Automated Vulnerability Scans

Manual

- ➡ Audit Log Generation
- Technical Documentation
- Mandatory SDL Training
- Architecture Diagram
- ➡ Attack Surface Analysis
- Access Control Matrix

Security Controls: Low

Secure Scaffolding

- Template for the project with secure defaults
- Standardization of integrated components, i.e., user management, session management
- Raise the bar

Automated Vulnerability Scans

- Establish automated system scans
- Integrate results in centralized system
- Track history and check for differences

Security Controls: Low

<u>Audit Log Generation</u>

- Create log output for application usage
- Focus on secure-critical functions
- Aggregate events in flows

<u>Attack Surface Analysis</u>

- Collect exposed interfaces
- Identify possible targets
- Get in to the perspective of an attacker

ERNW
providing security.

Security Controls: Medium

Low | **Medium** | High

Automatable

- ➡ Continuous Delivery Pipeline
- Application Security Scans
- Security Tests
- Regression Tests
- Robustness Tests
- Log Output Visualization
- Audit Log Alerting

Manual

- Code Review
- Data Flow Diagram
- Rule Definition
- Pair Programming
- ➡ Continuous Threat Modelling

Security Controls: Medium

Continuous Delivery Pipeline

- Deploy automatically to DEV/QS, manual to PROD
- Full access to DEV, limited Access to QS
- No PROD access et al., only to log sink

Continuous Threat Modelling

- Continuously feature delivery leads to continuously feature extension
- Identify new threats
- Document identified attack vectors, track them, and define mitigations

ERNW
providing security.

Security Controls: High

Low | Medium | **High**

Automatable

- ○ Performance Tests
- ➡ Regression Tests for Security Issues
- ○ Visualize Security Testing Results

Manual

- ○ Write Abuse Stories
- ○ External Code Review
- ➡ Mandatory Penetration Test
- ○ Data Format Definition
- ○ Decommissioning Concept
- ➡ Minion Penetration Tester

Security Controls: High

Regression Tests for Security Issues

- Establish regression tests for identified and resolved security issues
- Perform and monitor regression tests on regular base
- Track which modifications lead to unintended behaviors

Mandatory Penetration Test

- Establish process for external security verification
- Impersonate a real threat actor
- Track results and assign responsibilities

Security Controls: High

Minion Penetration Tester

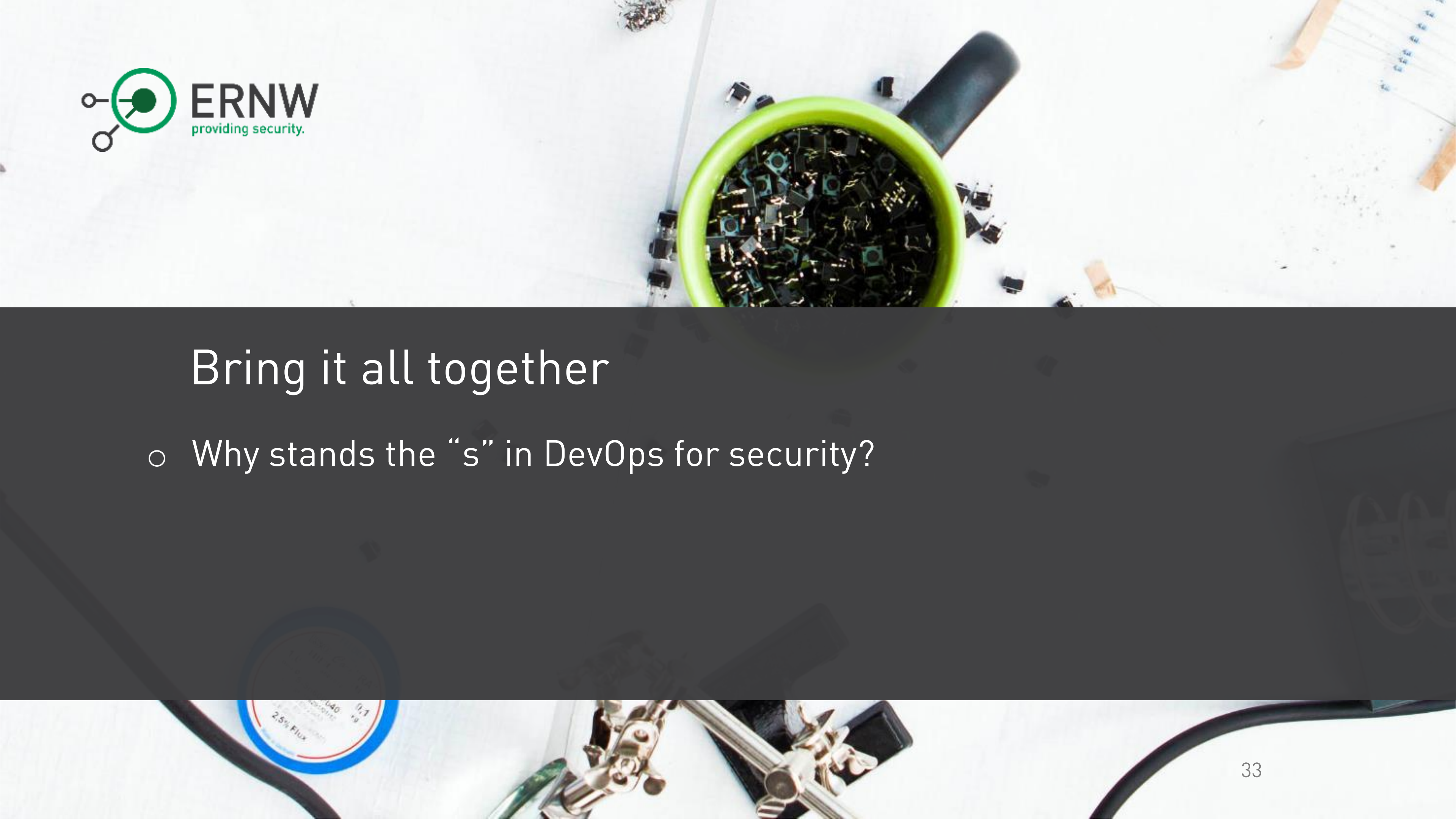
- Parallel with sprints
- Tests all new implemented features
- Sparring partner for security considerations

Video source: https://9gag.com/gag/aGZOemw/s

ERNW
providing security.

Security Design Principles

- Minimize the Attack Surface Area
- Establish Secure Defaults
- Least Privilege
- Defense in Depth
- Fail Securely
- Don’t Trust other Assets
- Separation of Duties
- Avoid Security by Obscurity
- Keep System-Architecture Simple
- Fix Security Issues Correctly



Bring it all together

- Why stands the “s” in DevOps for security?

Conclusion

- Individual implementation leads to individual issues
- Standardization and secure defaults raise the bar
- High rate of automation leverages direct and indirect benefits by transparency, speed, and reproducibility
- Early establishment of security leads to long-term cost reduction

ERNW
providing security.

Thank you for your Attention

Questions?

✉ jharrie@ernw.de

@NodyTweet
@WEareTROOPERS

www.ernw.de

www.insinuator.net

Sources

[1] HACKER-POWERED SECURITY REPORT 2019
https://www.hackerone.com/sites/default/files/2019-08/hacker-powered-security-report-2019.pdf

[2] THE STATE OF CROWDSOURCED SECURITY IN 2019
https://www.bugcrowd.com/resources/reports/priority-one-report/

[3] OWASP TOP10 2017
https://github.com/OWASP/Top10/blob/master/2017/OWASP%20Top%2010-2017%20(en).pdf

[4] Manifesto https://agilemanifesto.org/

[5] DevSecOps MM
https://www.owasp.org/index.php/OWASP_DevSecOps_Maturity_Model

[6] MS SDL https://www.microsoft.com/en-us/securityengineering/sdl/

[7] OSSTMM http://www.isecom.org/research/

[8] ISO https://www.iso.org/isoiec-27001-information-security.html

[9]OWASP TP10 2013
https://www.owasp.org/index.php/Category:OWASP_Top_Ten_2013_Project