# The Impact of Software Security Practices on Development Effort

## Sponsor: OUSD(R&E) | CCDC

**By**
**Elaine Venson**
**7th Annual SERC Doctoral Students Forum**
**November 18, 2019**
**FHI 360 CONFERENCE CENTER**
**1825 Connecticut Avenue NW, 8th Floor**
**Washington, DC 20009**

**www.sercuarc.org**

# Outline

MOTIVATION

SYSTEMATIC MAPPING

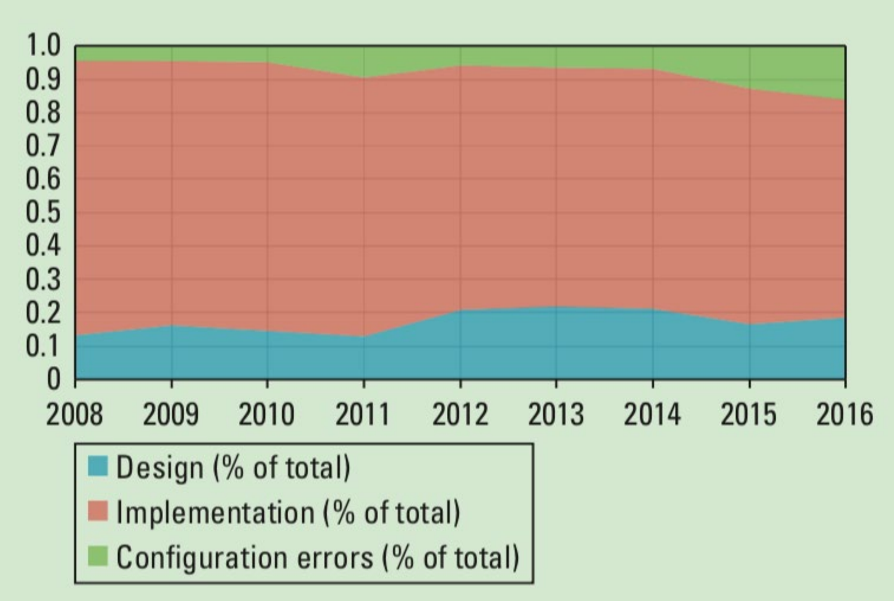
RESEARCH GOALS

SURVEY DESIGN

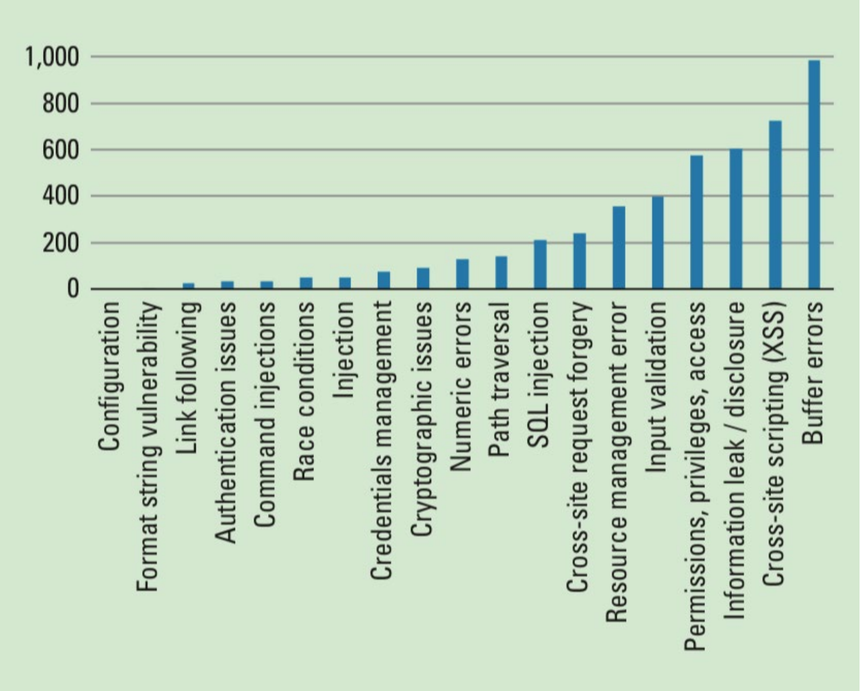
RESULTS

CONCLUSION

# Vulnerability Analysis



Based on the US National Vulnerabilities DB (NVD)
More than 85K publicly reported vulnerabilities



Distribution of Vulnerabilities in 2015
93% of buffer errors involved only a single condition
(typically, failure to check array bounds)

*Kuhn, M. Raunak, and R. Kacker, "It Doesn't Have to Be Like This: Cybersecurity Vulnerability Trends," IT Professional, vol. 19, no. 6, pp. 66–70, Nov. 2017.*

# Software Security

Engineering software that continues working under malicious attack [McGraw, 2004].

Many issues faced in computer security today are rooted in our approach to developing software and systems [Heitzenrater, 2016].

Software defects have security ramifications.

Security is an emergent property of a software system.

There is no single addition that can make a software secure.

# Costs to Find and Fix Defects

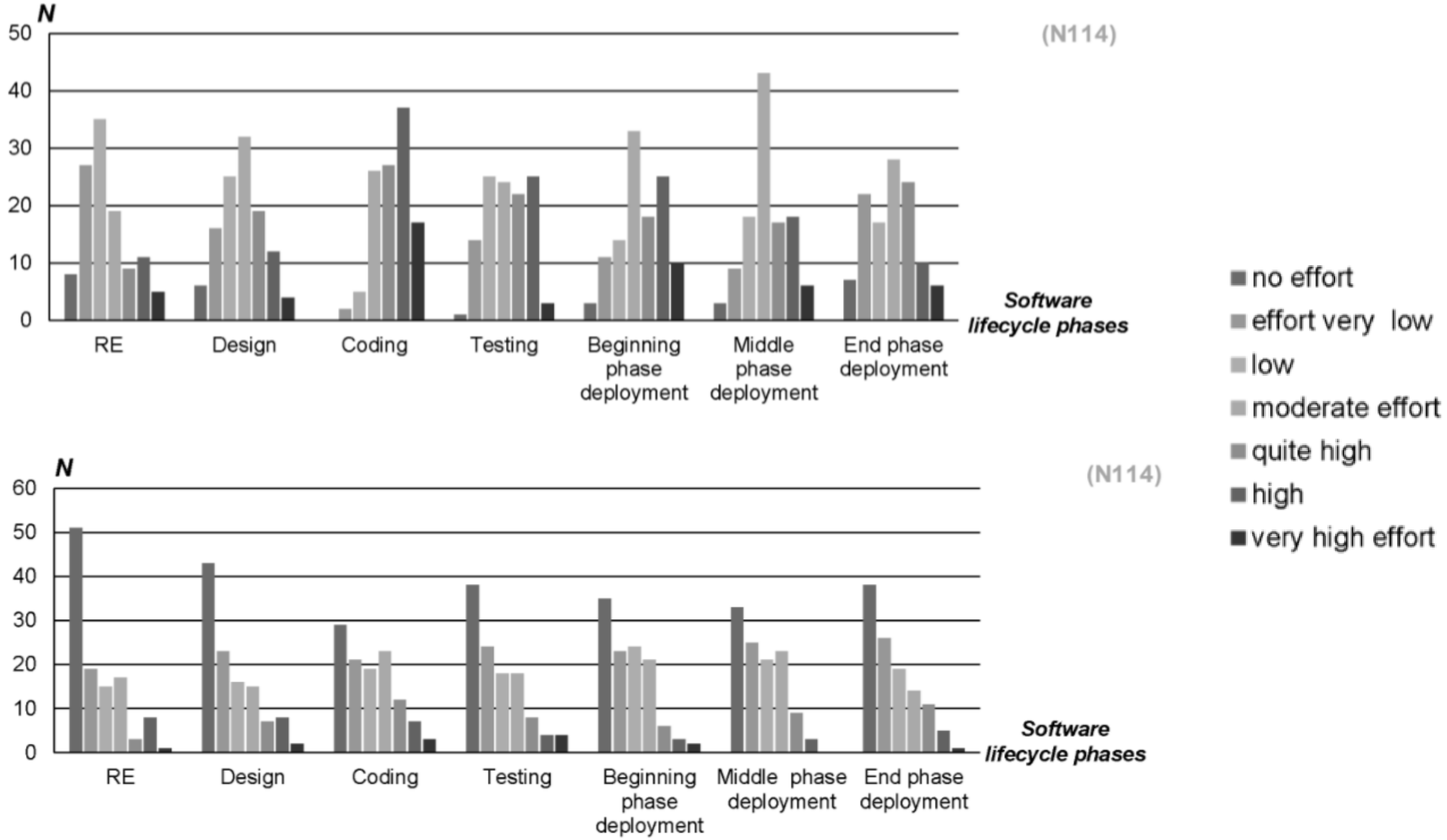
- Finding and fixing *non-severe* software defects after delivery is about ***twice as expensive*** as finding these defects pre-delivery.
- Finding and fixing a *severe* software problem after delivery is ***often 100 times more expensive*** than finding and fixing it during the requirements and design phase.

*Shull, F., Basili, V., Boehm, B., Brown, A.W., Costa, P., Lindvall, M., Port, D., Rus, I., Tesoriero, R., Zelkowitz, M., 2002. What we have learned about fighting defects.*

# General Development Effort x Security Effort

General Effort

Security Effort



*Chehrazi, G., Heimbach, I., Hinz, O.: The Impact of Security by Design on the Success of Open Source Software. In: ECIS 2016 Proceedings. p. 18 (2016).*

# Challenges

The **effort/costs** of performing security practices are often pointed out as a barrier to their wide use.

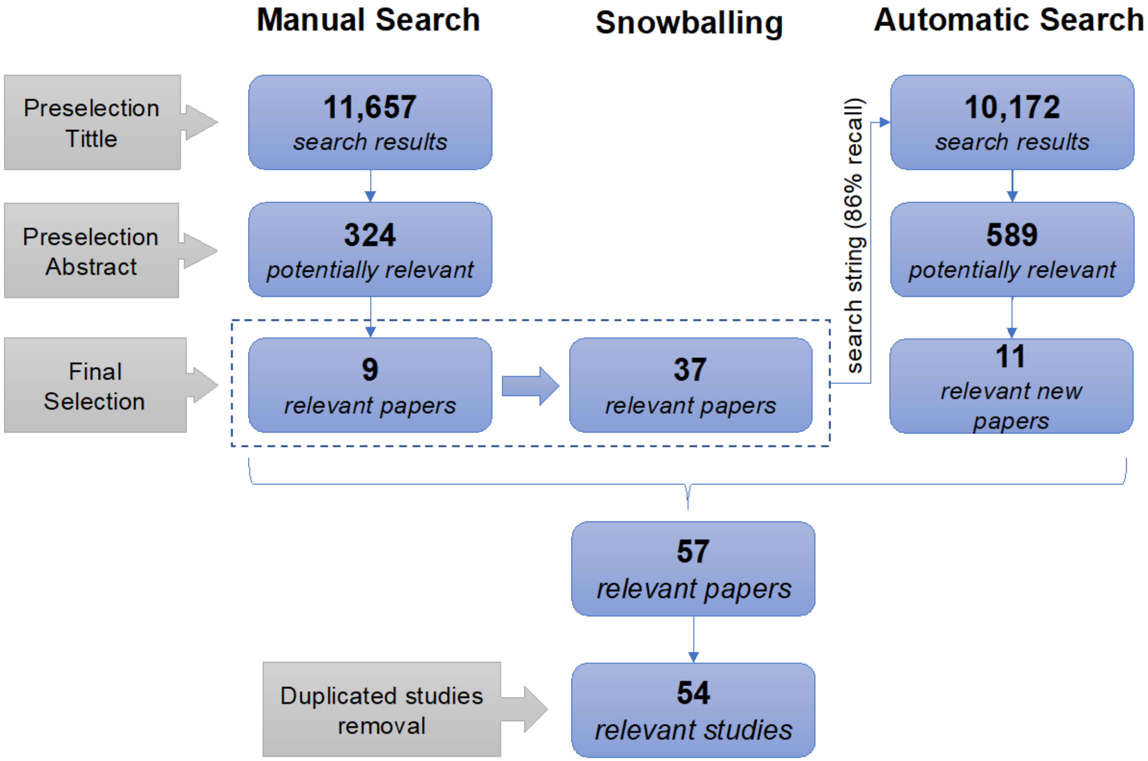
**Lack of knowledge** about the amount of resources needed to achieve a determined level of security assurance.

It is paramount for users, developers and managers to **understand and agree** on the right amount of resources to be allocated for software projects to deliver proper security.

# Systematic Mapping of Literature

**Inclusion Criteria:**

- IC1 – Study about software security that considers effort/cost impacts.
- IC2 – Study about effort/cost estimation or measurement that considers software security issues.



# Sources of Cost

| Source | Papers | Source | Papers |
|---|---|---|---|
| Perform Security Review | 21 | Perform Security Training | 6 |
| Apply Threat Modeling | 18 | Improve Development Process | 5 |
| Perform Security Testing | 16 | Perform Penetration Testing | 5 |
| Apply Security Requirements | 11 | Achieve Security Level | 3 |
| Apply Security Tooling | 11 | Document Technical Stack | 3 |
| Implement Countermeasures | 9 | Security Experts, Security Groups, Security Master | 3 |
| Fix Vulnerabilities | 9 | Track Vulnerabilities | 3 |
| Apply Secure Coding Standards | 8 | Functional Features | 2 |
| Apply Data Classifications Scheme | 7 | Hardening Procedures | 2 |
| Publish Operations Guide | 7 | Security by Design Paradigm | 1 |

# Approaches to Estimating Costs of SWSec

| Approach | Additional Cost | Source | Validation |
|---|---|---|---|
| COCOMO II security extension | 0.94 (Low)<br>1.02 (Nominal)<br>1.27 (High)<br>1.43 (Very High)<br>1.75 (Extra High) | Expert estimation | Not validated |
| COSECMO | 0% (Nominal)<br>20% to 80% (EAL 3 - High)<br>50 to 200% (EAL 4 - Very High)<br>125% to 500% (EAL 5 - Extra High)<br>313% to 1250% (EAL 6 - Super High)<br>781% to 3125% (EAL 7 - Ultra High) | Expert estimation | Not validated |
| Weapon systems development cost model (COCOMO II based) | 1.0 (Low or Nominal)<br>1.87 (High) | Expert estimation and 73 data points | Cross validation |
| Secure OS software cost model (COCOMO II based) | 1 (Nominal)<br>1.25 to 1.5 (High)<br>1.75 to 2.0 (Very High)<br>2.0 to 2.75 (Extra High)<br>3.0 to 3.75 (Super High) | Expert estimation | Case study |
| FPA security extension | 0 to 5% increase in the function points size of the project | Practices from survey with developers | Not validated |

# Research Goals

Gather a better understating of how software security practices are applied in the industry.

- Effort and frequency of activities.

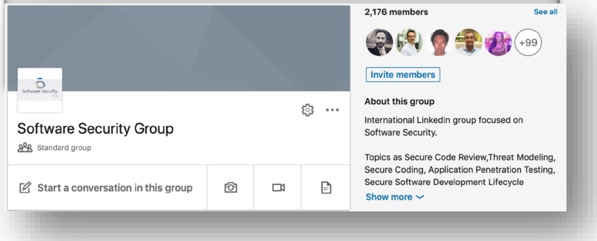
Identify the implications of applying such activities in terms of effort.

- Effort added in projects.
- Effort estimation methods.

# Survey Design

## Sampling Frame

- *Software Security Group on LinkedIn*
- 2012 member at the time



Software Security Group

2,176 members

About this group

International LinkedIn group focused on Software Security.

Topics as Secure Code Review,Threat Modeling, Secure Coding, Application Penetration Testing, Secure Software Development Lifecycle

## Sampling Strategy

- *Random Sampling*
- *Initial sample size = 908*
- *Excluding recruiters and sales people = 808*

## Recruitment Strategy

- *Manual invitation through LinkedIn messages*
- *Raffle on Amazon to encourage responses*

## Questionnaire Design

- *Reviewed by external expert*
- *Piloted with 10 members from the sampling frame*
- *Close-ended and quantitative questions*
- *One open-ended questions*

## Data Collection and Analysis

- *Web-based tool*
- *Available for 2 weeks*
- *Reminder after 1 week*
- *Quantitative analysis mostly*

# Results

110 complete responses

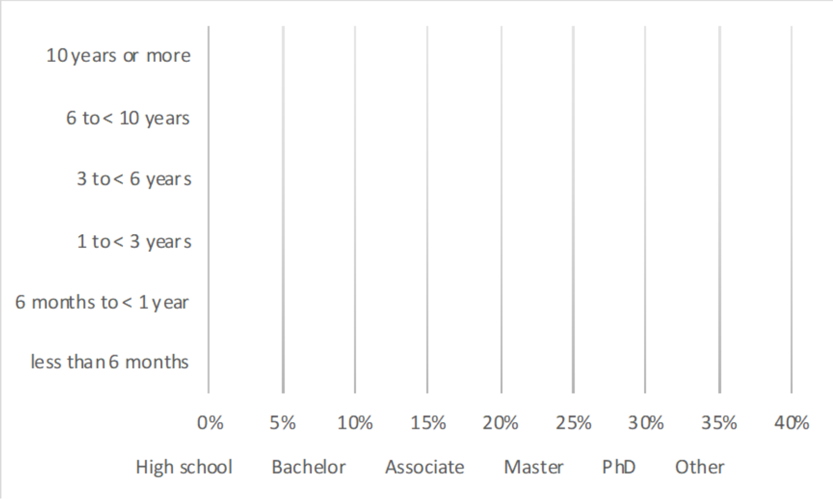
13.61%
of the sample

Confidence Interval
9.07

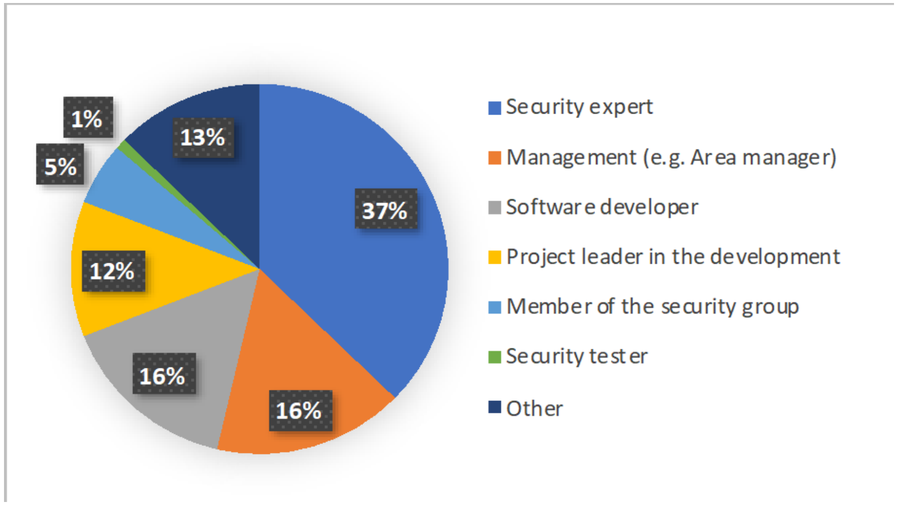
Level of Confidence
95%

# Participants Background

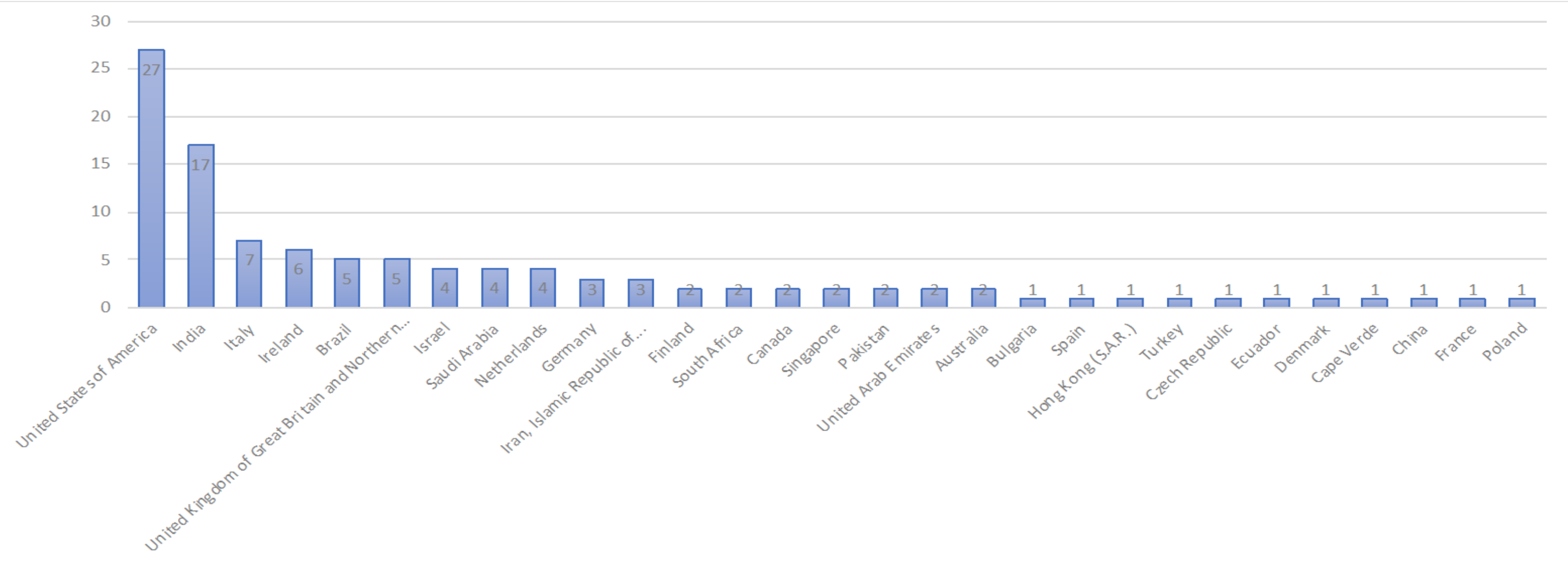
## Experience and Degree



## Position in Organization

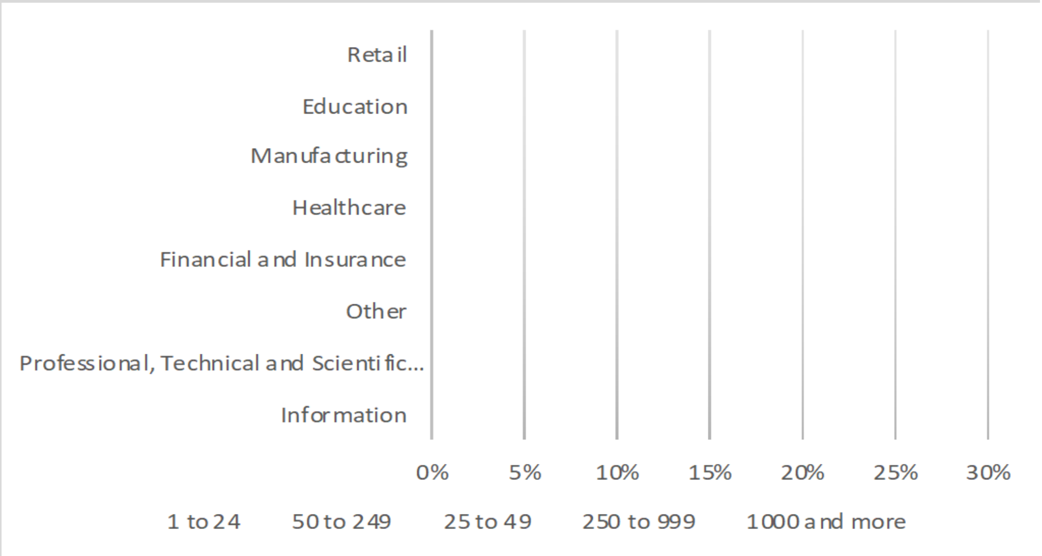


## Countries

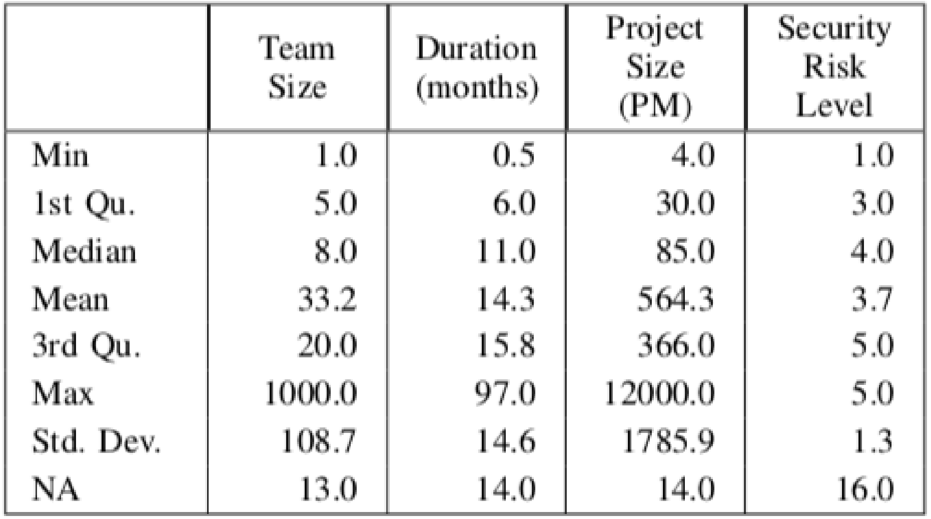


# Organization and Projects

## Organization Size and Domain

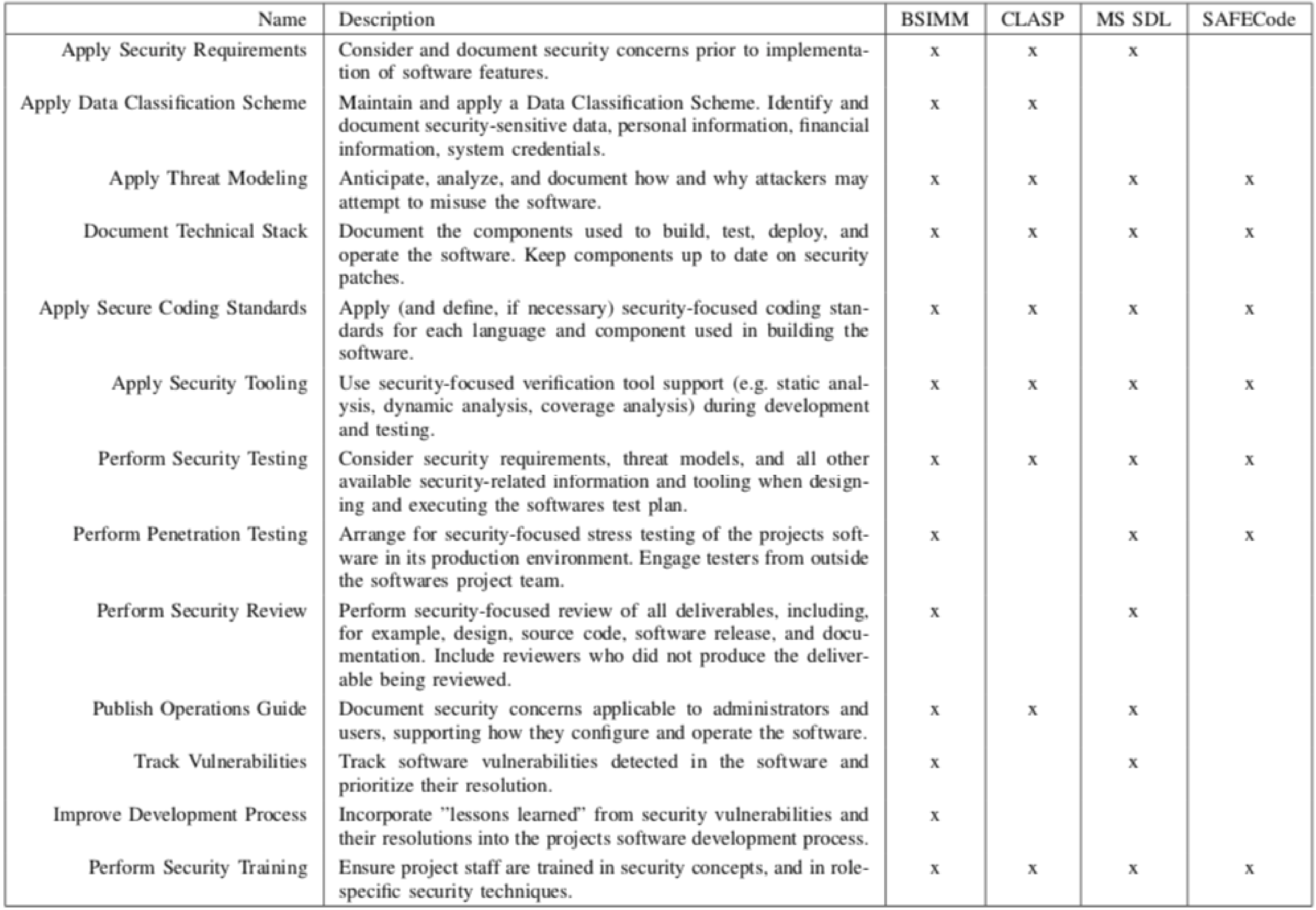


## Selected Project



| | Team Size | Duration (months) | Project Size (PM) | Security Risk Level |
|---|---|---|---|---|
| Min | 1.0 | 0.5 | 4.0 | 1.0 |
| 1st Qu. | 5.0 | 6.0 | 30.0 | 3.0 |
| Median | 8.0 | 11.0 | 85.0 | 4.0 |
| Mean | 33.2 | 14.3 | 564.3 | 3.7 |
| 3rd Qu. | 20.0 | 15.8 | 366.0 | 5.0 |
| Max | 1000.0 | 97.0 | 12000.0 | 5.0 |
| Std. Dev. | 108.7 | 14.6 | 1785.9 | 1.3 |
| NA | 13.0 | 14.0 | 14.0 | 16.0 |

# Software Security Practices

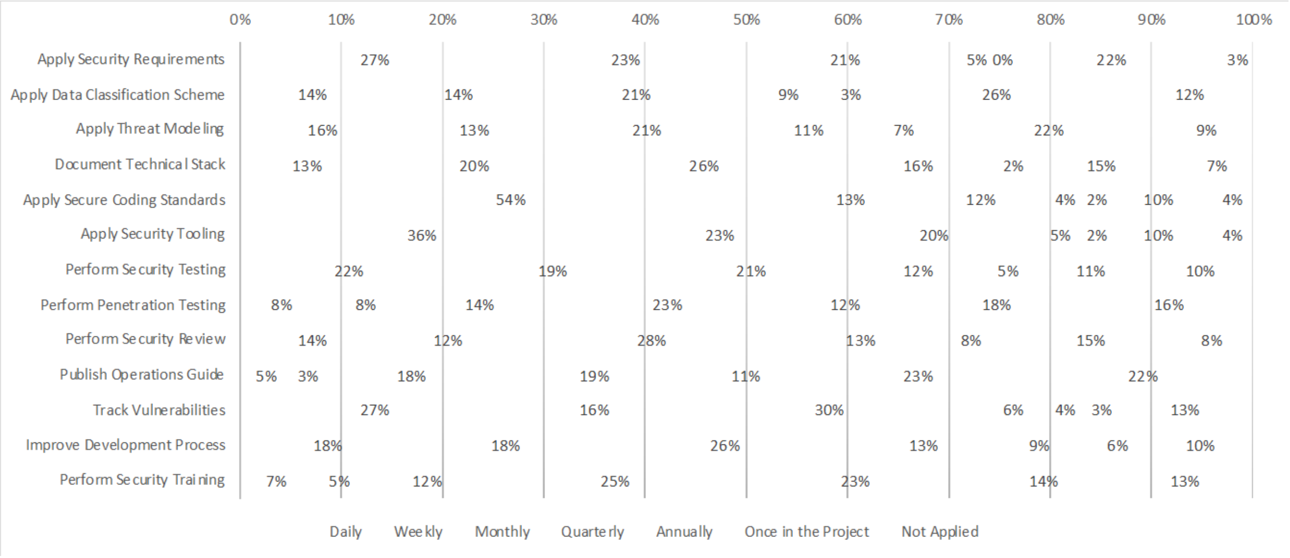


| Name | Description | BSIMM | CLASP | MS SDL | SAFECode |
|---|---|---|---|---|---|
| Apply Security Requirements | Consider and document security concerns prior to implementation of software features. | x | x | x | |
| Apply Data Classification Scheme | Maintain and apply a Data Classification Scheme. Identify and document security-sensitive data, personal information, financial information, system credentials. | x | x | | |
| Apply Threat Modeling | Anticipate, analyze, and document how and why attackers may attempt to misuse the software. | x | x | x | x |
| Document Technical Stack | Document the components used to build, test, deploy, and operate the software. Keep components up to date on security patches. | x | x | x | x |
| Apply Secure Coding Standards | Apply (and define, if necessary) security-focused coding standards for each language and component used in building the software. | x | x | x | x |
| Apply Security Tooling | Use security-focused verification tool support (e.g. static analysis, dynamic analysis, coverage analysis) during development and testing. | x | x | x | x |
| Perform Security Testing | Consider security requirements, threat models, and all other available security-related information and tooling when designing and executing the softwares test plan. | x | x | x | x |
| Perform Penetration Testing | Arrange for security-focused stress testing of the projects software in its production environment. Engage testers from outside the softwares project team. | x | | x | x |
| Perform Security Review | Perform security-focused review of all deliverables, including, for example, design, source code, software release, and documentation. Include reviewers who did not produce the deliverable being reviewed. | x | | x | |
| Publish Operations Guide | Document security concerns applicable to administrators and users, supporting how they configure and operate the software. | x | x | x | |
| Track Vulnerabilities | Track software vulnerabilities detected in the software and prioritize their resolution. | x | | x | |
| Improve Development Process | Incorporate ”lessons learned” from security vulnerabilities and their resolutions into the projects software development process. | x | | | |
| Perform Security Training | Ensure project staff are trained in security concepts, and in role-specific security techniques. | x | x | x | x |

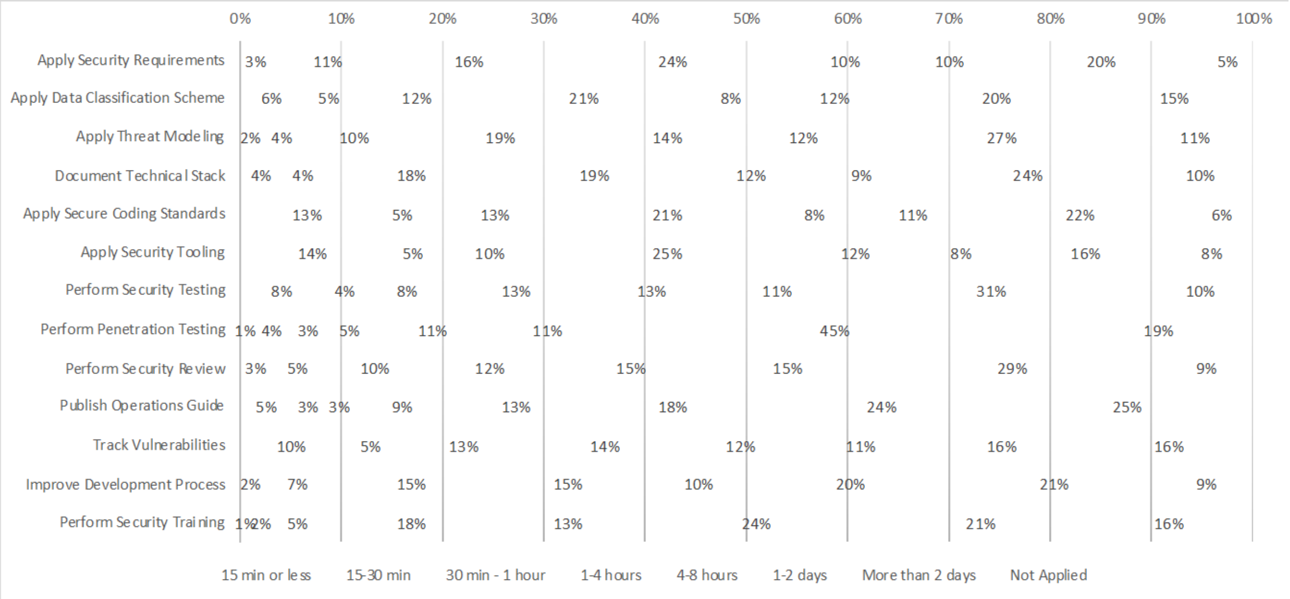
Morrison, P., Smith, B.H., Williams, L., 2017. Surveying Security Practice Adherence in Software Development, in: Proceedings of the Hot Topics in Science of Security: Symposium and Bootcamp, HoTSoS. ACM, New York, NY, USA, pp. 85–94.

# Practices Frequency and Effort

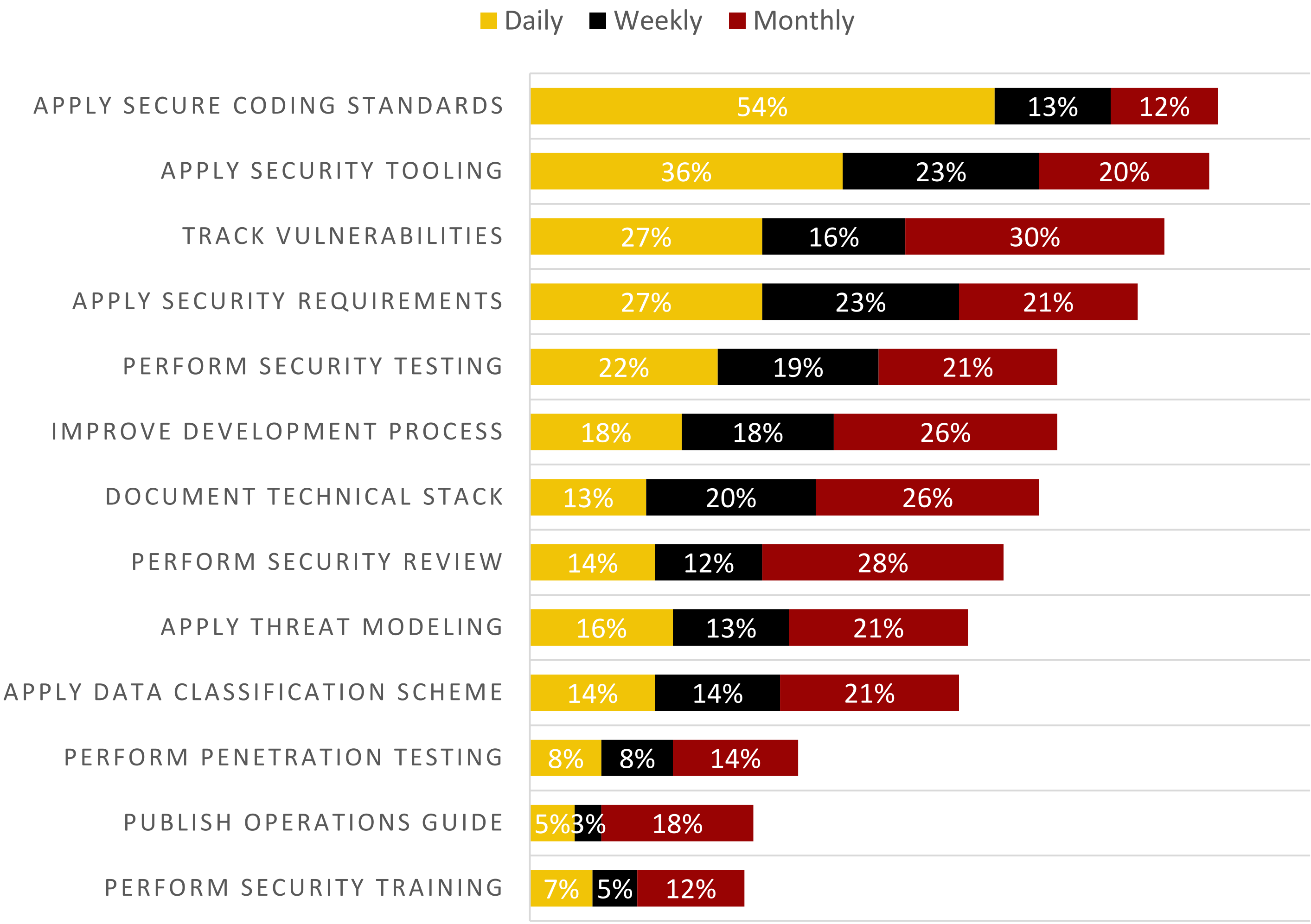
## Frequency of Application



## Effort Each Application

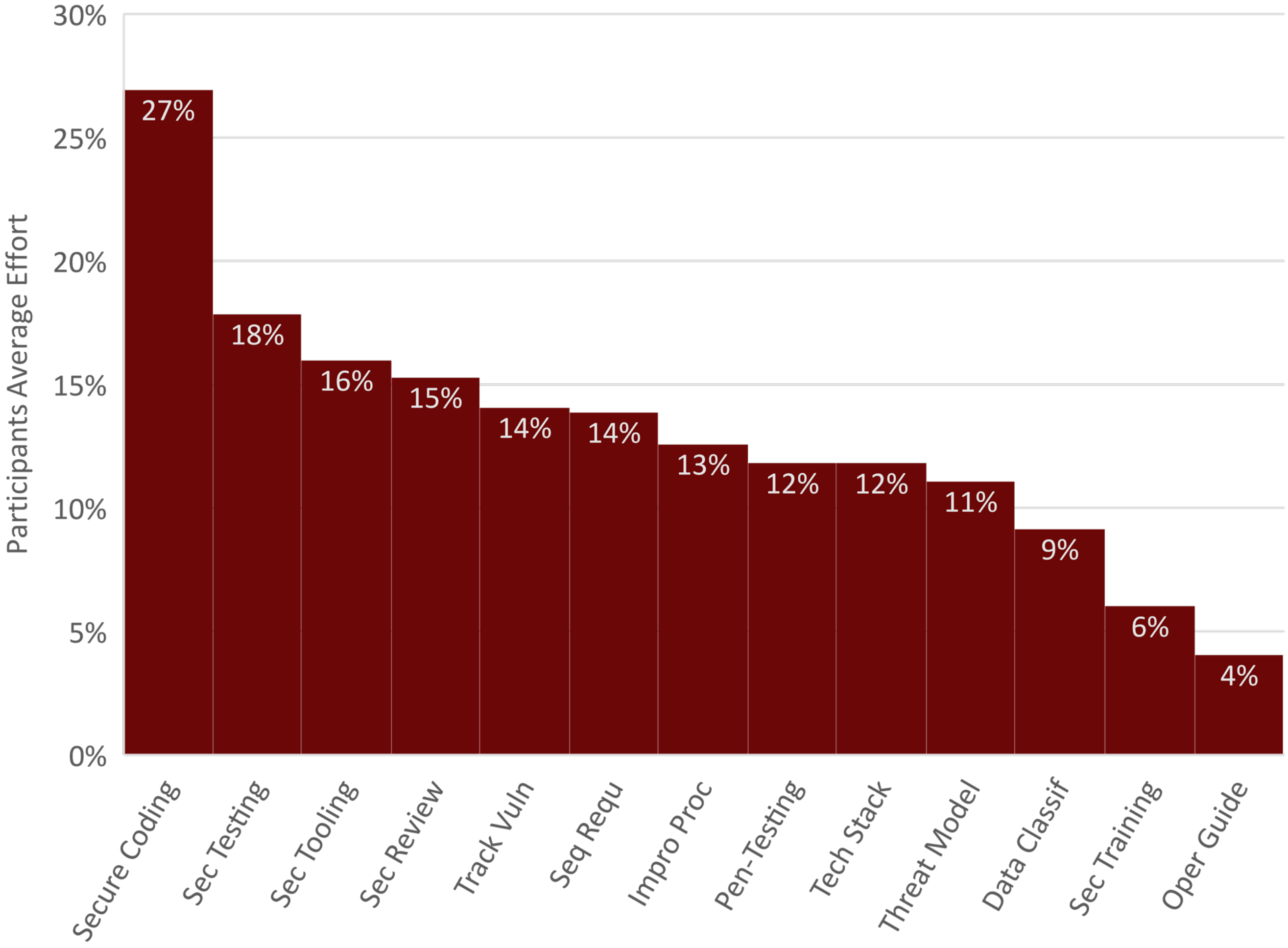


# Most Often Executed Practices



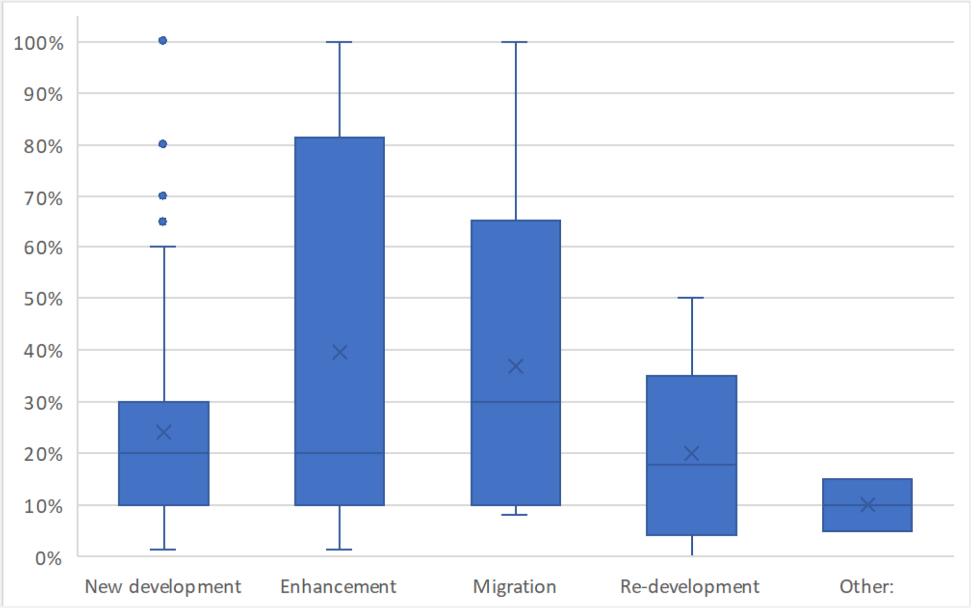
# % Individual Effort on Security Practices

(Frequency x Effort each application) / One-person project effort

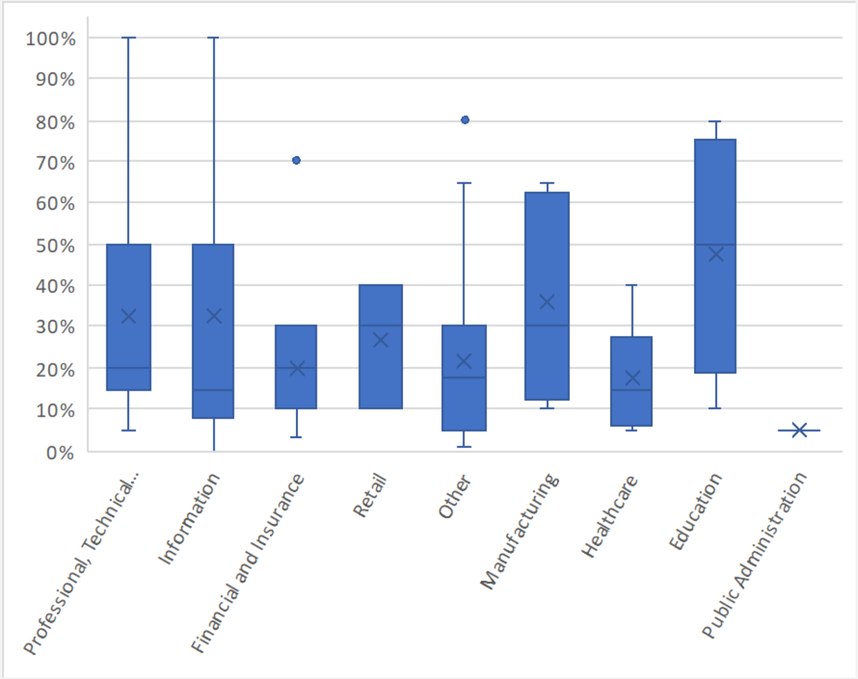


# Effort Dedicated to Security

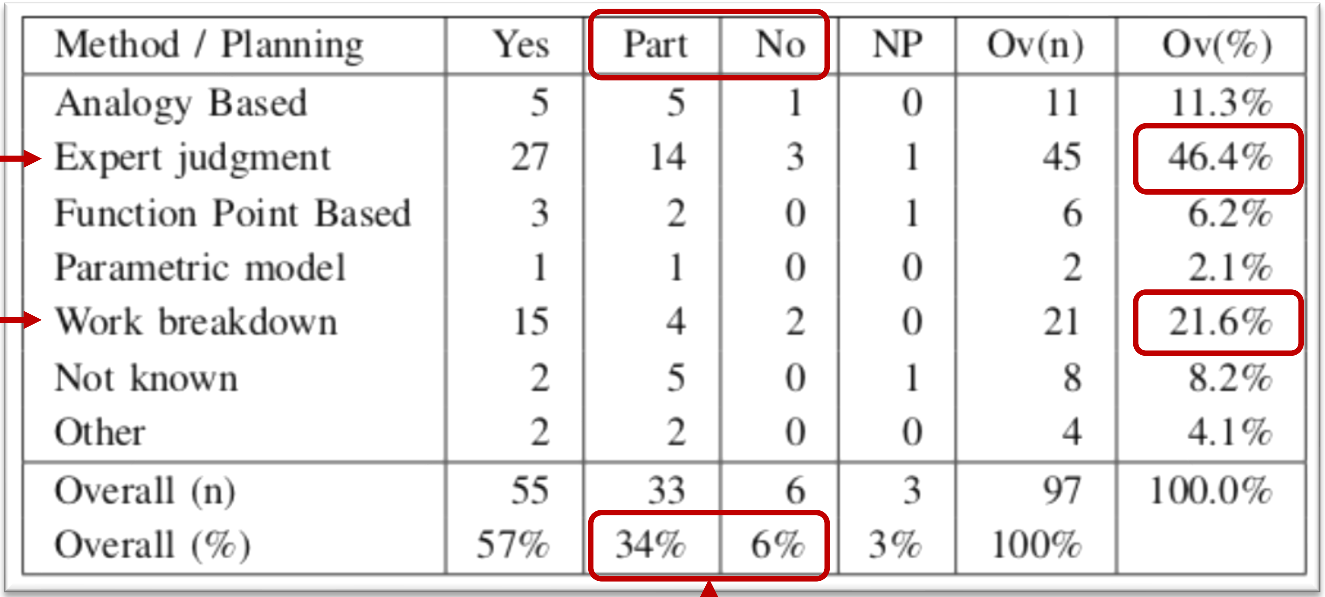
## By Development Type



## By Sector



# Estimation Methods and Use



| Method / Planning | Yes | Part | No | NP | Ov(n) | Ov(%) |
|---|---|---|---|---|---|---|
| Analogy Based | 5 | 5 | 1 | 0 | 11 | 11.3% |
| Expert judgment | 27 | 14 | 3 | 1 | 45 | 46.4% |
| Function Point Based | 3 | 2 | 0 | 1 | 6 | 6.2% |
| Parametric model | 1 | 1 | 0 | 0 | 2 | 2.1% |
| Work breakdown | 15 | 4 | 2 | 0 | 21 | 21.6% |
| Not known | 2 | 5 | 0 | 1 | 8 | 8.2% |
| Other | 2 | 2 | 0 | 0 | 4 | 4.1% |
| Overall (n) | 55 | 33 | 6 | 3 | 97 | 100.0% |
| Overall (%) | 57% | 34% | 6% | 3% | 100% | |

Practices were partially or not planned.

# Challenges in Estimating/Planning Security Practices

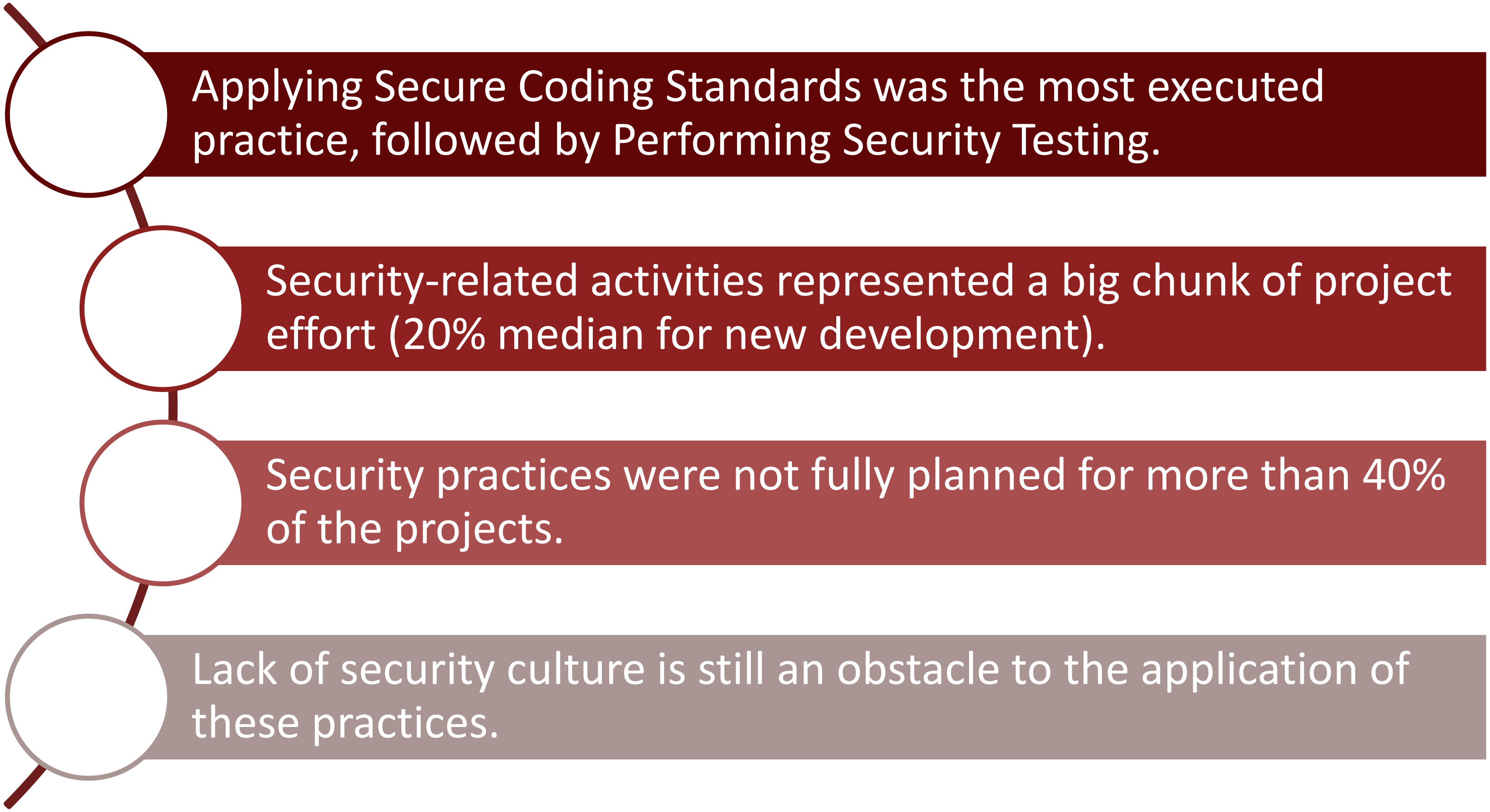
## Lack of security culture from developers, managers and business stakeholders

- *"There are a few, but getting people to truly stop, and understand 100% why the best practices are needed, can be a challenge - when people get focused on delivery dates. Once you explain the 'What could happen...' - it tends to sink in."*
- *"Always people considered security as feature to add after business logic and programming are finished so it happens to delay the project a lot."*
- *"Convincing project manager to incorporate security related time and effort."*
- *"Low priority from higher management, strict delivery deadlines - all estimates were hard or rejected."*

## Prioritization of business features upon security

- *"Business wants least time in security as the delivery is (the) main focus."*
- *"Fast development, to get feature out. Feature priority, security takes back seat sometimes."*
- *"Estimating time/effort wasn't the real challenge. It was more of getting a buy-in from Development team regarding time allocation for security assurance activities as these were generally given lower priority due to their non-functional nature compared to business/functional tasks."*

# Conclusion



- Applying Secure Coding Standards was the most executed practice, followed by Performing Security Testing.
- Security-related activities represented a big chunk of project effort (20% median for new development).
- Security practices were not fully planned for more than 40% of the projects.
- Lack of security culture is still an obstacle to the application of these practices.

# References

- Chehrazi, G., Heimbach, I., Hinz, O., 2016. The Impact of Security by Design on the Success of Open Source Software, in: ECIS 2016 Proceedings. Presented at the European Conference on Information Systems (ECIS), p. 18.
- Kuhn, R., Raunak, M., Kacker, R., 2017. It Doesn’t Have to Be Like This: Cybersecurity Vulnerability Trends. IT Professional 19, 66–70. https://doi.org/10.1109/MITP.2017.4241462
- Shull, F., Basili, V., Boehm, B., Brown, A.W., Costa, P., Lindvall, M., Port, D., Rus, I., Tesoriero, R., Zelkowitz, M., 2002. What we have learned about fighting defects, in: Proceedings Eighth IEEE Symposium on Software Metrics. Presented at the Proceedings Eighth IEEE Symposium on Software Metrics, pp. 249–258. https://doi.org/10.1109/METRIC.2002.1011343
- Venson, E., Guo, X., Yan, Z., Boehm, B., 2019. Costing Secure Software Development: A Systematic Mapping Study, in: Proceedings of the 14th International Conference on Availability, Reliability and Security, ARES ’19. ACM, New York, NY, USA, pp. 9:1–9:11. https://doi.org/10.1145/3339252.3339263
- Venson, E., Alfayez, R., Marília M. F., G., Rejane M. C., F., Boehm, B., 2019. The Impact of Software Security Practices on Development Effort: An Initial Survey, in: 2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). Presented at the 2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 1–12. https://doi.org/10.1109/ESEM.2019.8870153

# Thank you!

## The Impact of Software Security Practices on Development Effort

An Initial Survey

*Elaine Venson*
*venson@usc.edu*

*Paper Authors:*
*Elaine Venson, Reem Alfayez, Marília M. F. Gomes, Rejane M. C. Figueiredo, Barry Boehm*