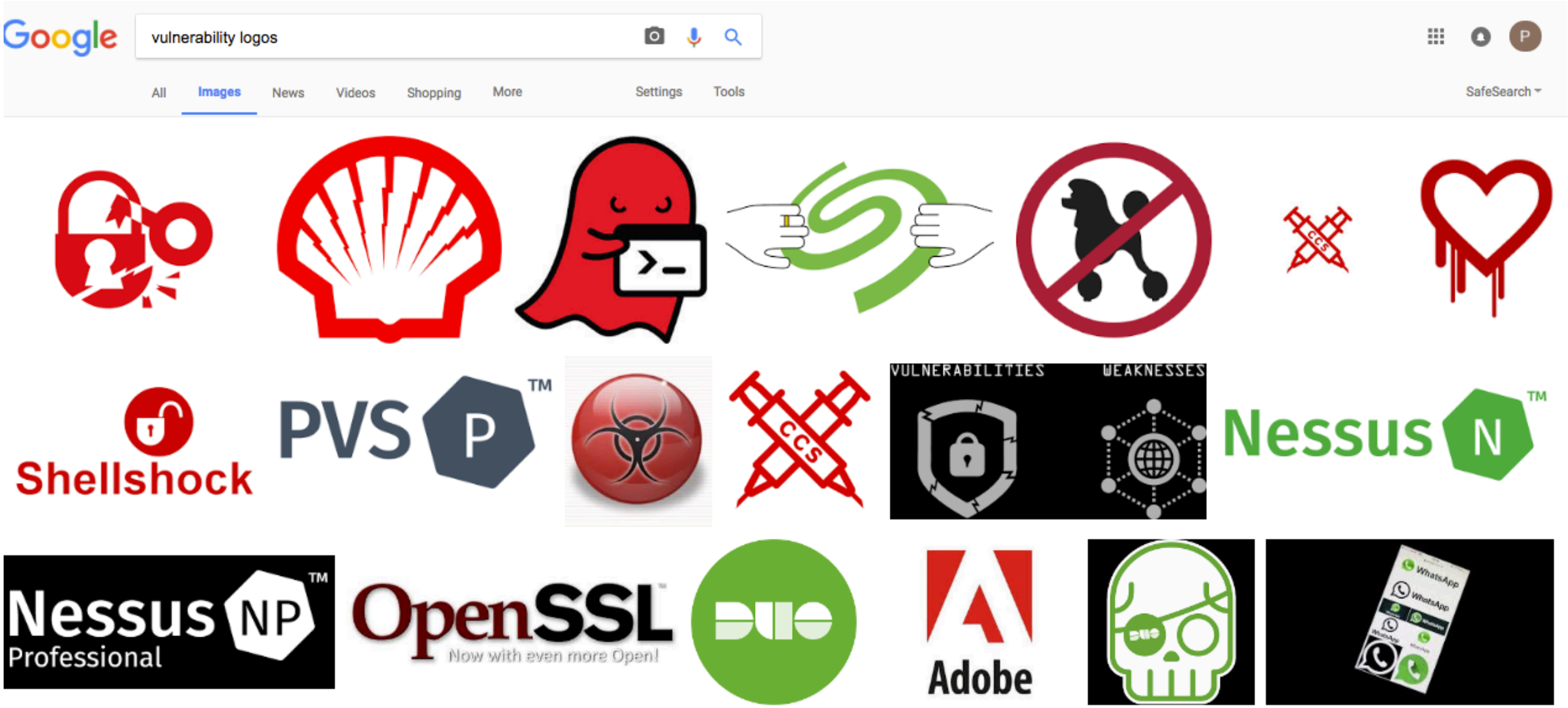
Surveying Security Practice Adherence in Software Development

Patrick Morrison, Laurie Williams

North Carolina State University

Ben Smith, IBM, Inc.


Google
vulnerability logos
All
Images
News
Videos
Shopping
More
Settings
Tools
SafeSearch
P
Shellshock
PVS
P
TM
CCS
VULNERABILITIES
WEAKNESSES
Nessus
N
TM
Nessus
NP
TM
Professional
OpenSSL
Now with even more Open!
DUO
Adobe
WhatsApp

http://cdn.makeuseof.com/wp-content/uploads/2014/09/stress-free-programming-frustration.jpg

112 Practices

61 Practices

SAFECode

Software Assurance Forum for Excellence in Code

Driving Security and Integrity

Microsoft

Security Development Lifecycle

184 Practices

OWASP

Open Web Application Security Project

41 Practices

www.bsimm.com www.safecode.org http://www.microsoft.com/en-us/sdl/ www.owasp.org

http://cdn.makeuseof.com/wp-content/uploads/2014/09/stress-free-programming-frustration.jpg

Goal

To support researcher investigations of software development security practice adherence by building and validating a set of security practices and adherence measures through literature review and survey data analysis

Research Questions

- RQ1: What software development security practices are used by software development teams?
- RQ2: Does security practice adherence, as measured by Ease of use, Effectiveness, and Training, correlate with software development security practice use?

RQ1: What practices?
Methods

- Literature review of four lists of recommended software development security practices (BSIMM, SDL, SAFECode, OWASP CLASP), to identify common security practices.
- Surveyed 11 open source projects based on the practices we identified.

RQ1: What practices? Classification

- For each security-related action described in one of our source documents, we:
  - Categorize the action taken as a **Practice**
  - Identify **Artifact Affected** by the action
  - Identify **Artifact Referenced** by the action
  - Identify the **Verb**
  - Identify **Role** of person to apply the practice
  - Identify the life cycle **Phase** during which the action takes place

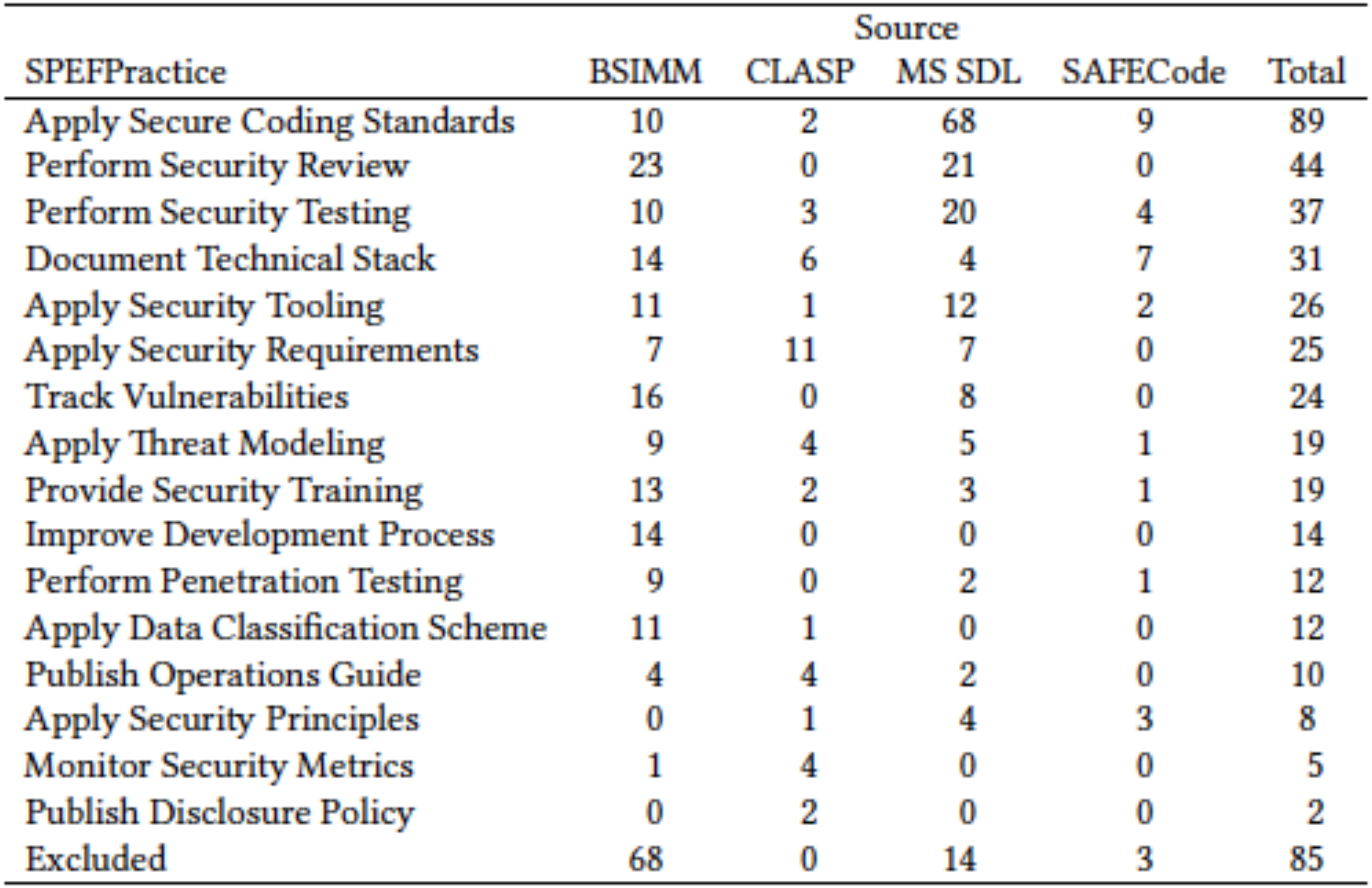
RQ1: What practices? Classification Example 1

- Practice: “Apply Secure Coding Standard”
  - BSIMM: “Use Coding Standards”
  - SAFECode: “Avoid String Concatenation for Dynamic SQL Statements”
  - MS SDL: “NULL out freed memory pointers in code”
  - **Artifact Affected:** Source Code
  - **Artifact Referenced:** Coding Standard
  - **Verb:** Apply
  - **Role**: Developer
  - **Phase**: Implementation

RQ1: What practices? Example 2 (Excluded practice)

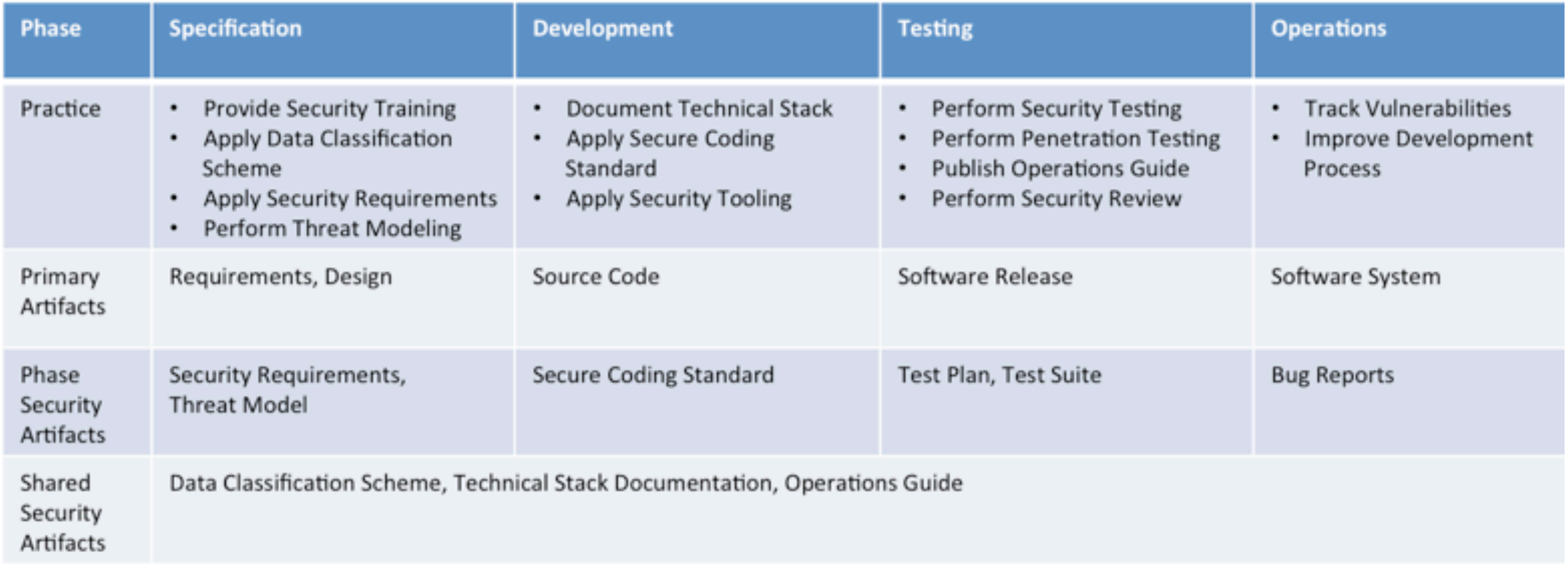
- Excluded: “Develop an operations inventory of applications” (BSIMM)
  - **Artifact Affected:** Operations Inventory (Application Portfolio)
  - **Artifact Referenced:** Software Applications
  - **Verb:** Develop
  - **Role**: Manager, Project Manager
  - **Phase**: Operations

RQ1: What practices? – Classification Results



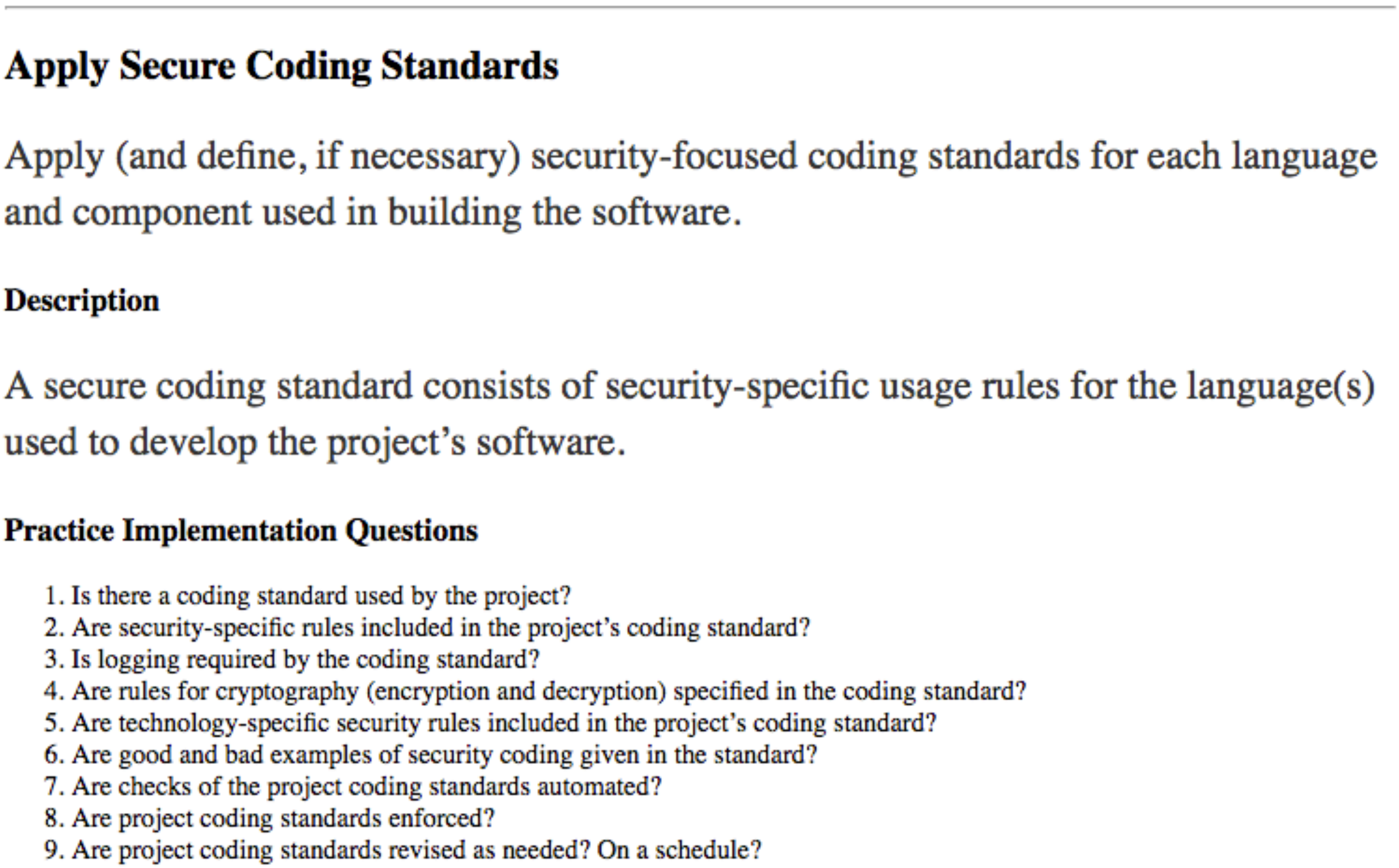
	Source				
SPEFPractice	BSIMM	CLASP	MS SDL	SAFECode	Total
Apply Secure Coding Standards	10	2	68	9	89
Perform Security Review	23	0	21	0	44
Perform Security Testing	10	3	20	4	37
Document Technical Stack	14	6	4	7	31
Apply Security Tooling	11	1	12	2	26
Apply Security Requirements	7	11	7	0	25
Track Vulnerabilities	16	0	8	0	24
Apply Threat Modeling	9	4	5	1	19
Provide Security Training	13	2	3	1	19
Improve Development Process	14	0	0	0	14
Perform Penetration Testing	9	0	2	1	12
Apply Data Classification Scheme	11	1	0	0	12
Publish Operations Guide	4	4	2	0	10
Apply Security Principles	0	1	4	3	8
Monitor Security Metrics	1	4	0	0	5
Publish Disclosure Policy	0	2	0	0	2
Excluded	68	0	14	3	85

Practices in the Software Life Cycle



Phase	Specification	Development	Testing	Operations
Practice	• Provide Security Training • Apply Data Classification Scheme • Apply Security Requirements • Perform Threat Modeling	• Document Technical Stack • Apply Secure Coding Standard • Apply Security Tooling	• Perform Security Testing • Perform Penetration Testing • Publish Operations Guide • Perform Security Review	• Track Vulnerabilities • Improve Development Process
Primary Artifacts	Requirements, Design	Source Code	Software Release	Software System
Phase Security Artifacts	Security Requirements, Threat Model	Secure Coding Standard	Test Plan, Test Suite	Bug Reports
Shared Security Artifacts	Data Classification Scheme, Technical Stack Documentation, Operations Guide			

Example Practice Definition



Apply Secure Coding Standards

Apply (and define, if necessary) security-focused coding standards for each language and component used in building the software.

Description

A secure coding standard consists of security-specific usage rules for the language(s) used to develop the project's software.

Practice Implementation Questions

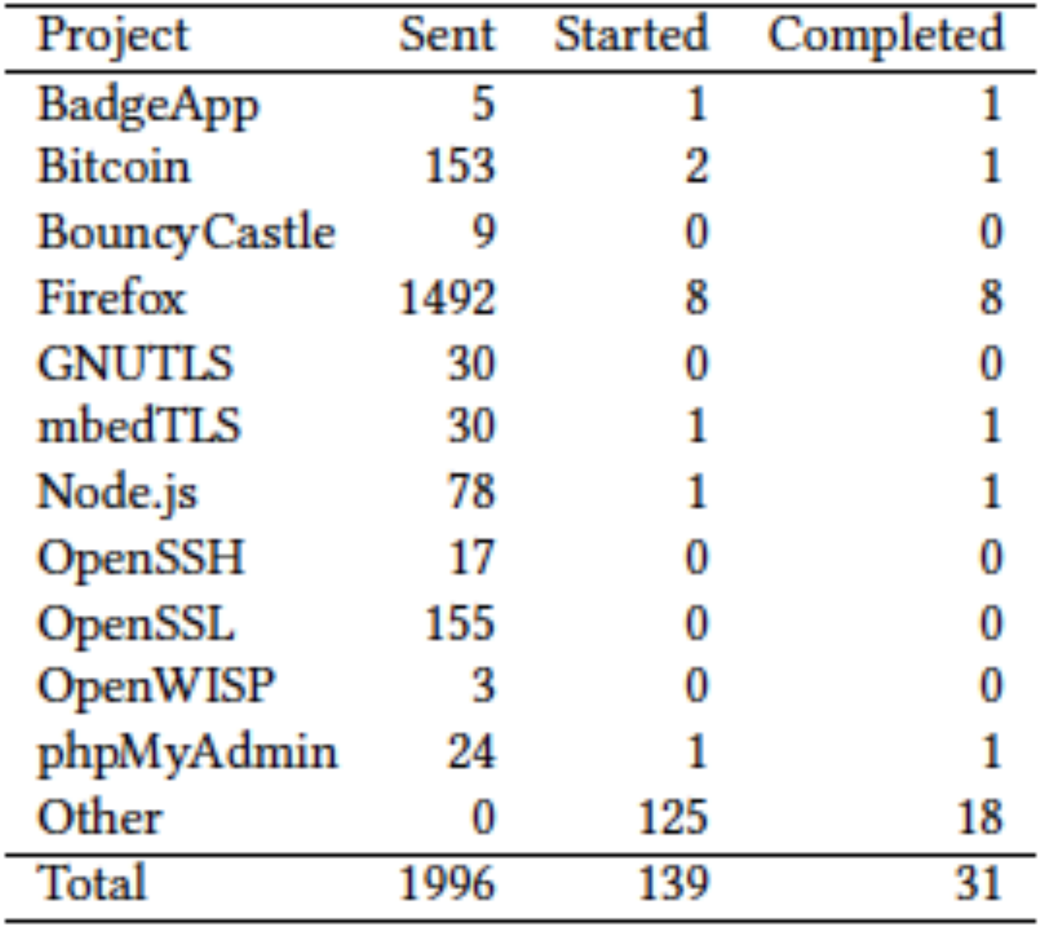
1. Is there a coding standard used by the project?
2. Are security-specific rules included in the project's coding standard?
3. Is logging required by the coding standard?
4. Are rules for cryptography (encryption and decryption) specified in the coding standard?
5. Are technology-specific security rules included in the project's coding standard?
6. Are good and bad examples of security coding given in the standard?
7. Are checks of the project coding standards automated?
8. Are project coding standards enforced?
9. Are project coding standards revised as needed? On a schedule?

- http://pjmorris.github.io/Security-Practices-Evaluation-Framework/guidebook.html

RQ1: What practices, **actually**? Survey

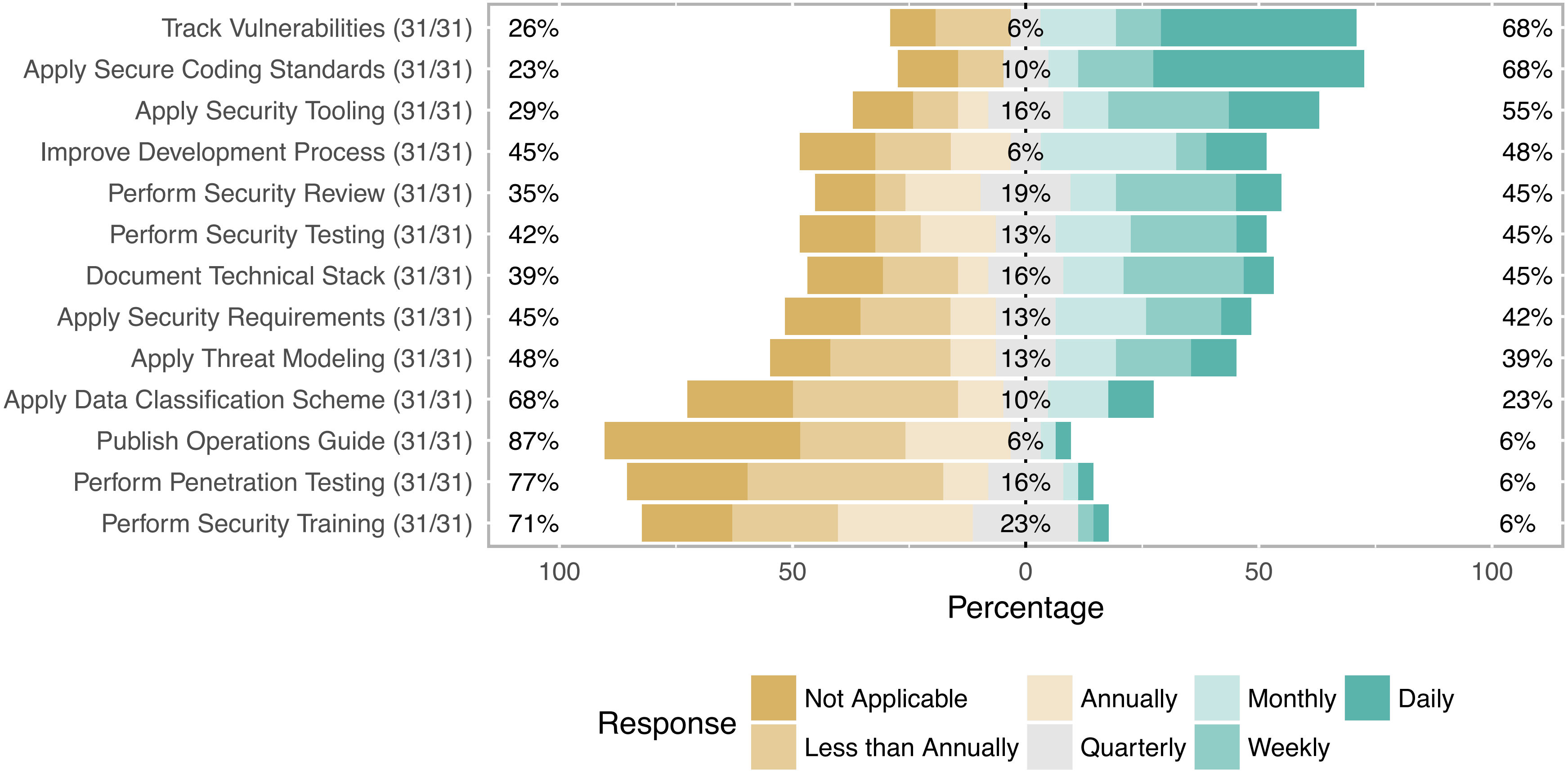
- How does our list compare with what software development teams **actually do** for the sake of security?
- Developed a survey instrument to collect security practice adherence data at the level of the software development team.
- Survey security-focused software development teams

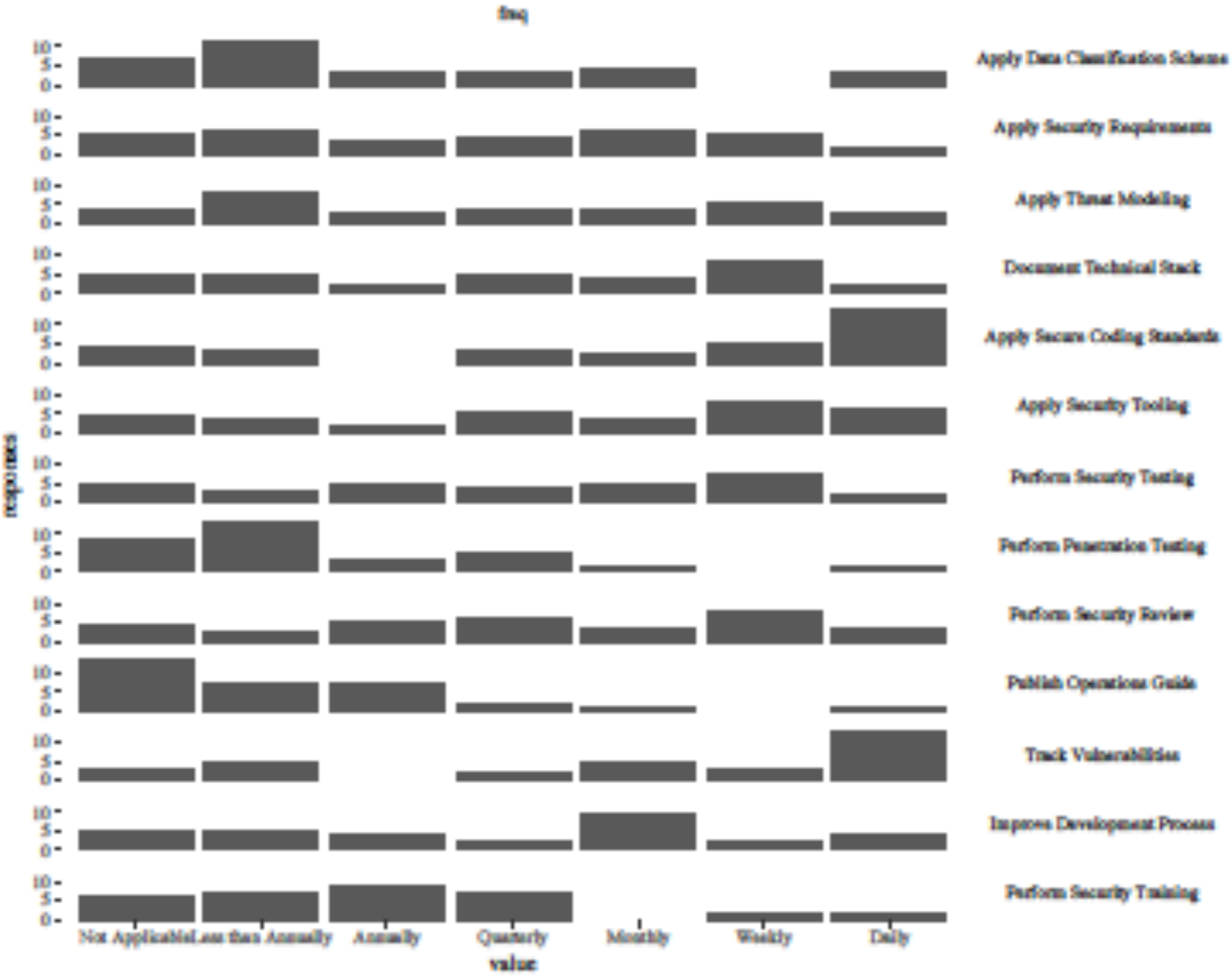
RQ1: What practices?
Survey Demographics



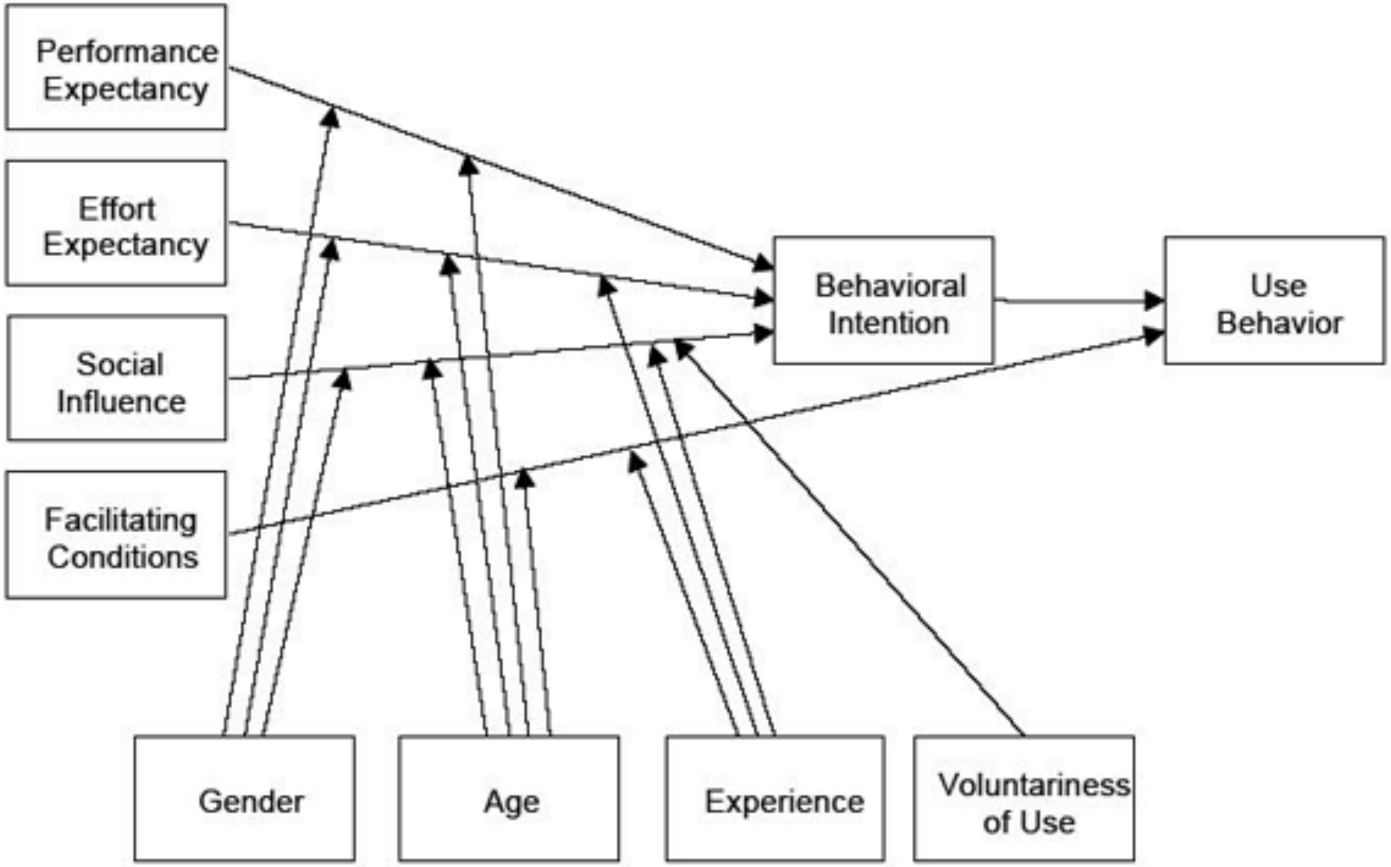
Project	Sent	Started	Completed
BadgeApp	5	1	1
Bitcoin	153	2	1
BouncyCastle	9	0	0
Firefox	1492	8	8
GNUTLS	30	0	0
mbedTLS	30	1	1
Node.js	78	1	1
OpenSSH	17	0	0
OpenSSL	155	0	0
OpenWISP	3	0	0
phpMyAdmin	24	1	1
Other	0	125	18
Total	1996	139	31

“How often do you engage in the following activities?”




freq
responses
value
10
5
0
Not Applicable
Less than Annually
Annually
Quarterly
Monthly
Weekly
Daily
Apply Data Classification Scheme
Apply Security Requirements
Apply Threat Modeling
Document Technical Stack
Apply Secure Coding Standards
Apply Security Tooling
Perform Security Testing
Perform Penetration Testing
Perform Security Review
Publish Operations Guide
Track Vulnerabilities
Improve Development Process
Perform Security Training

RQ2: Practice adherence? Theory



http://www.vvenkatesh.com/Research/IT%20Images/UTAUT.gif

Venkatesh, V., Morris M.G., Davis, F.D., and Davis, G.B.
“User Acceptance of Information Technology: Toward a Unified View,” MIS Quarterly, 27, 2003, 425-478

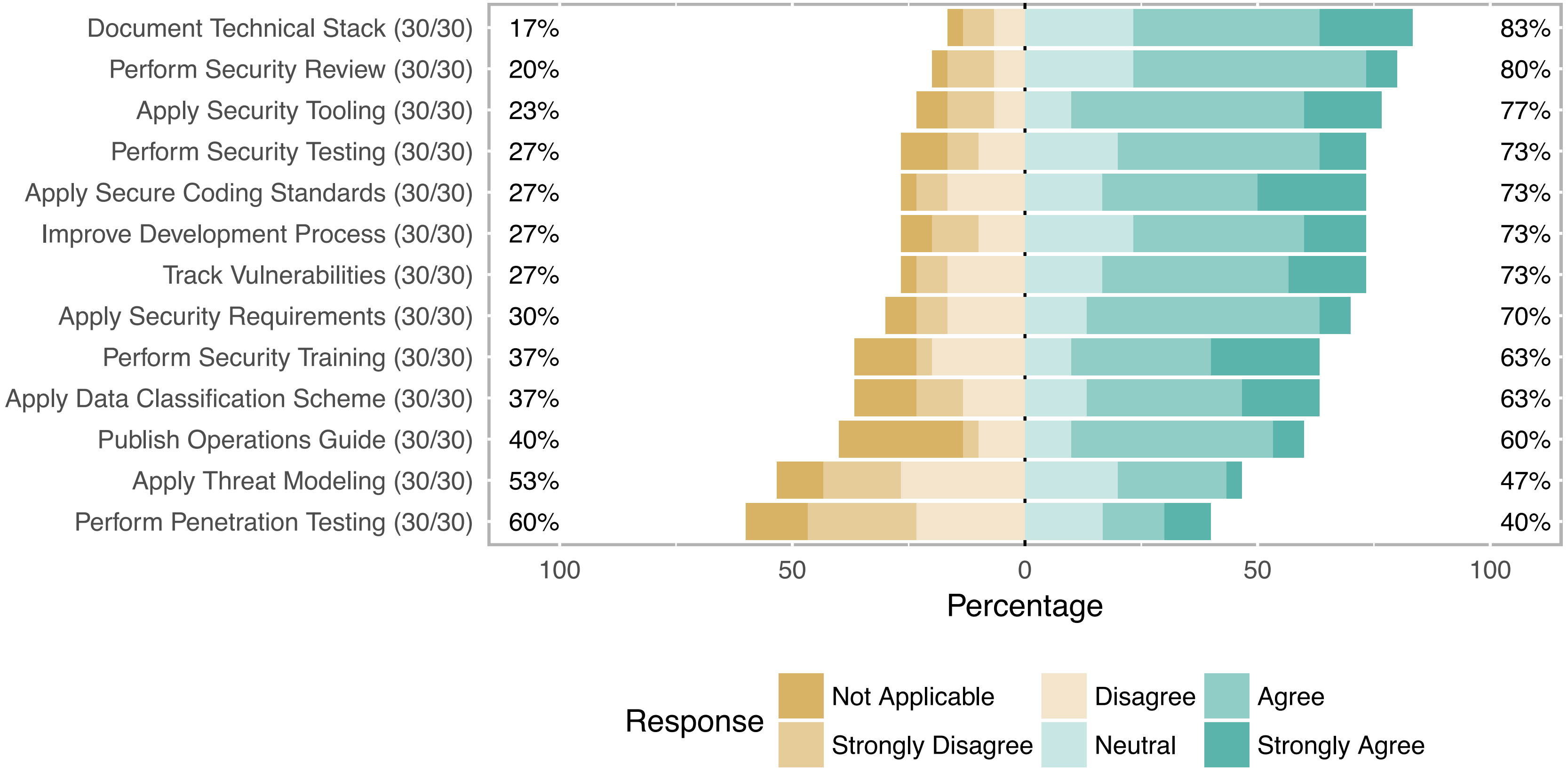
RQ2: Practice Adherence

- RQ2: Does security practice adherence, as measured by Ease of use, Effectiveness, and Training, correlate with software development security practice use?

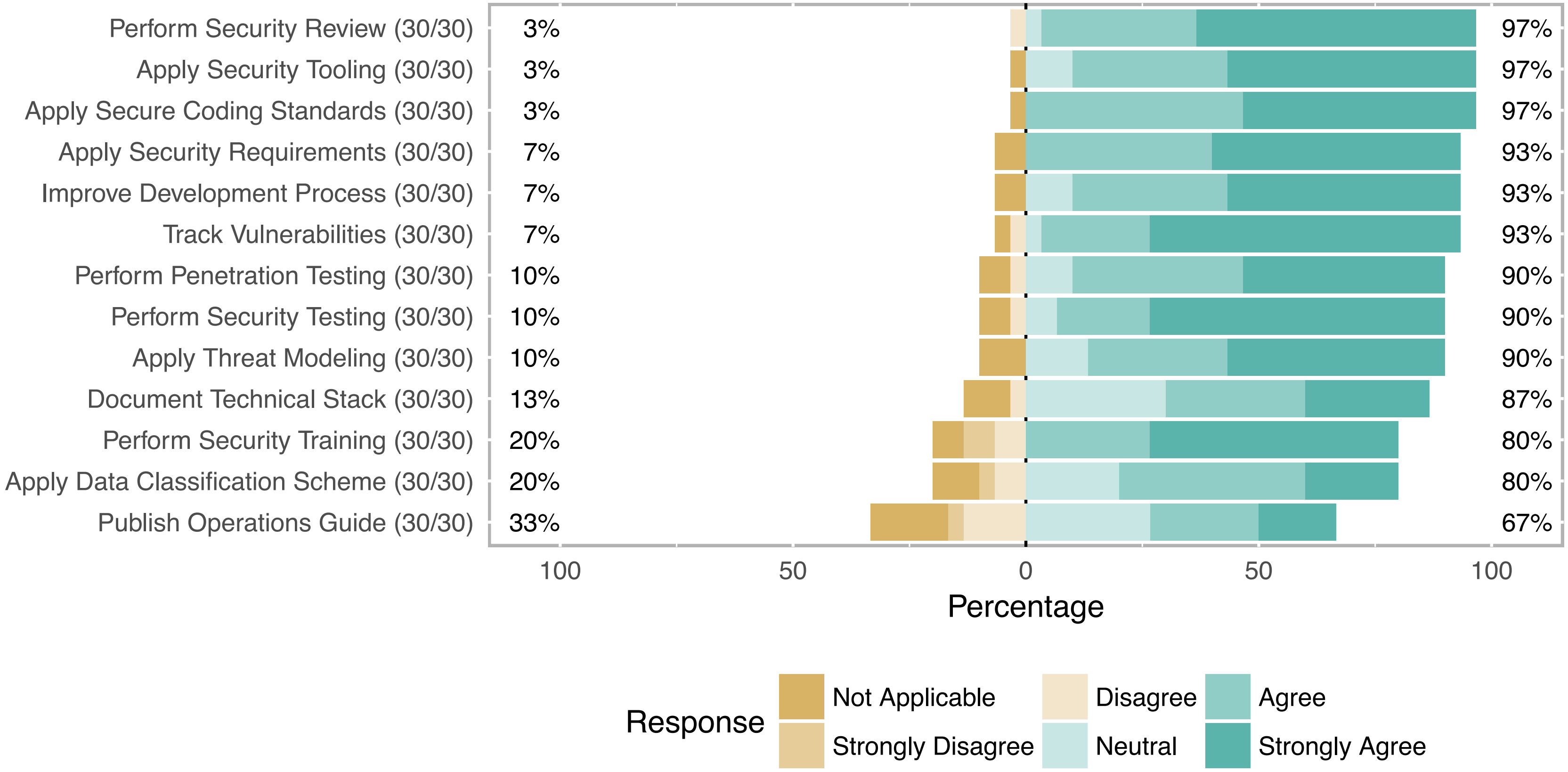
RQ2: Practice adherence? Hypotheses

- Ease of use affects frequency of use of software development security practices.
- Effectiveness affects frequency of use of software development security practices.
- Training affects frequency of use of software development security practices.

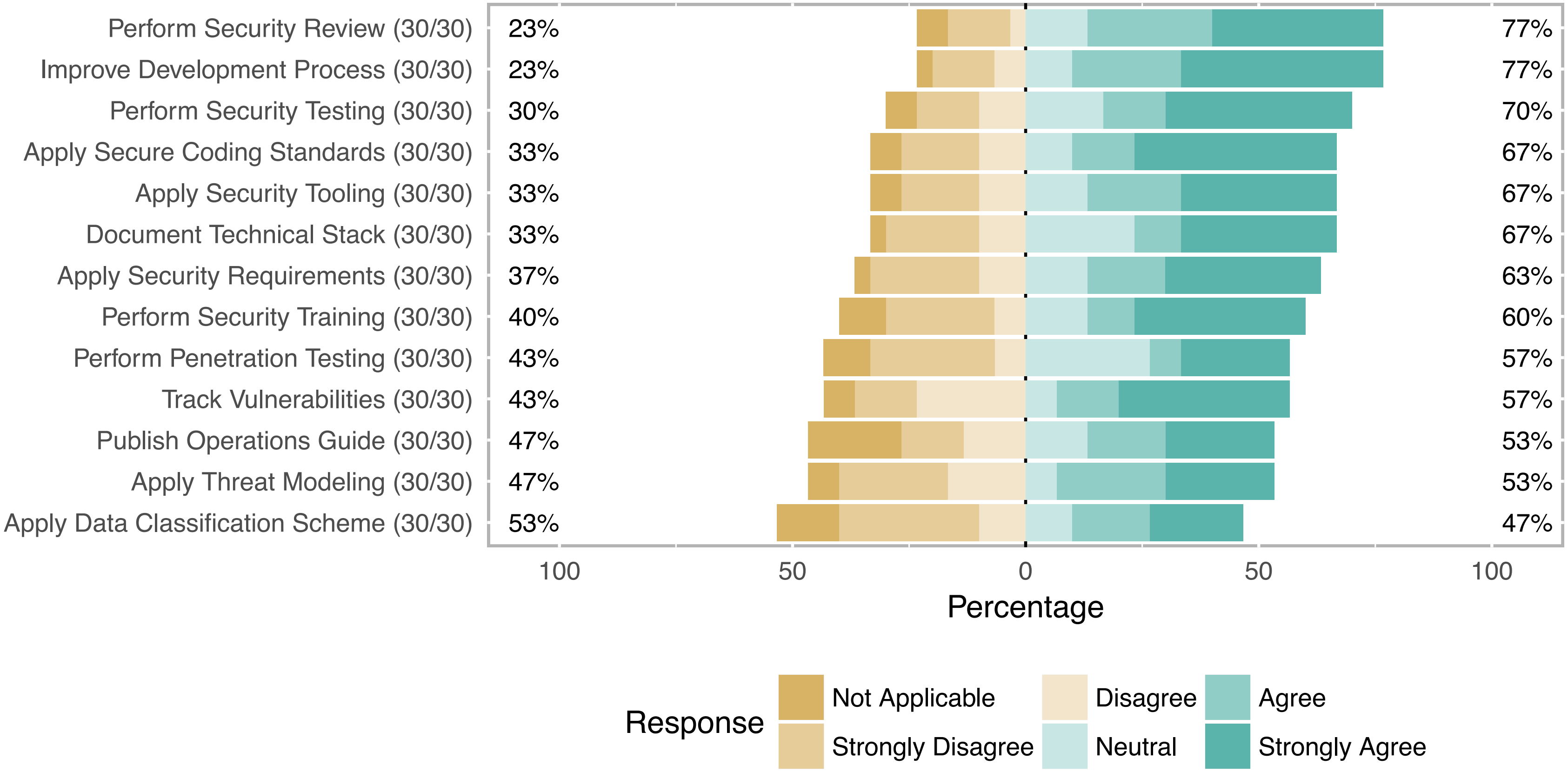
“This practice is easy to use”



"This practice assists in preventing and/or removing vulnerabilities"



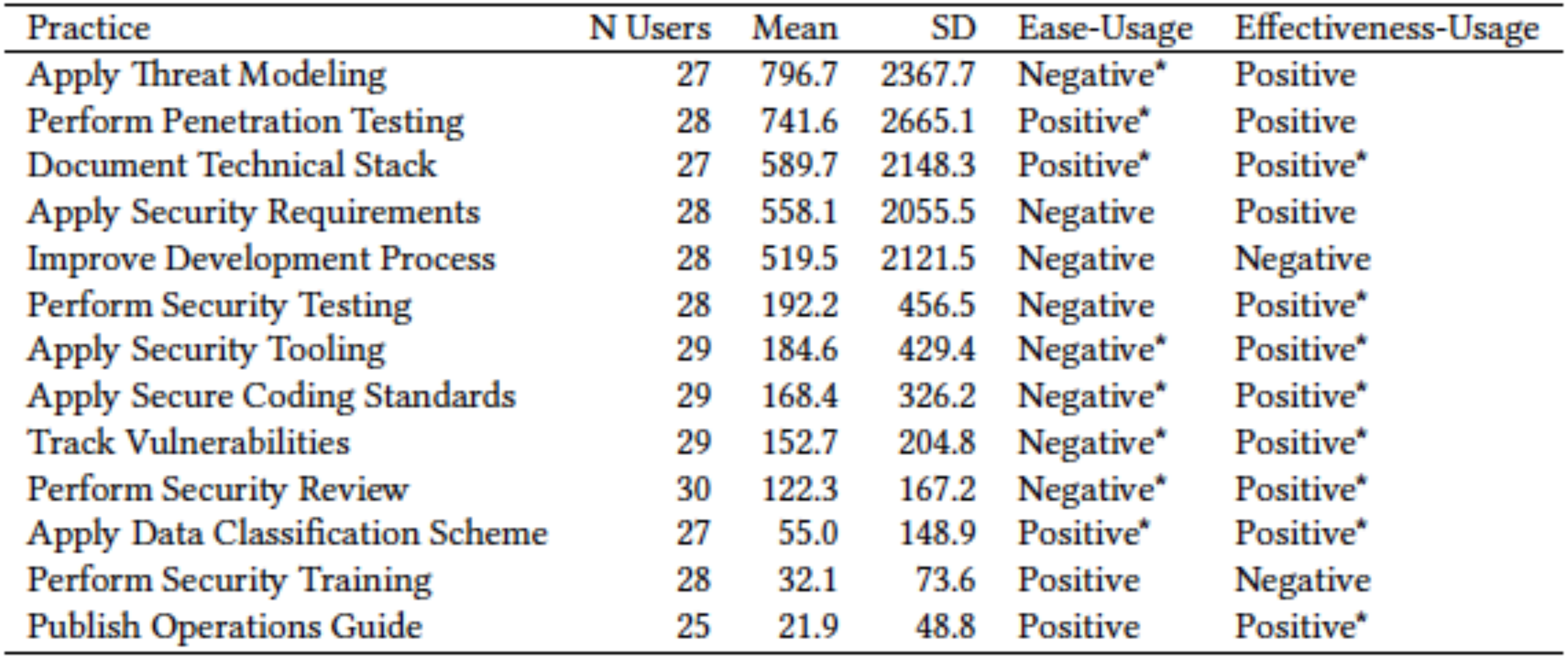
“I have been trained in the use of this practice”



“I have been trained in the use of this practice”

Survey respondent: "I would remove security training. Classroom knowledge delivered via lecture is useless at best. Experiential knowledge and mentorship through hands on experience is the only way to learn. Academics have trouble with this one, but it is true. Sorry.”

RQ2: Practice adherence?



Practice	N Users	Mean	SD	Ease-Usage	Effectiveness-Usage
Apply Threat Modeling	27	796.7	2367.7	Negative*	Positive
Perform Penetration Testing	28	741.6	2665.1	Positive*	Positive
Document Technical Stack	27	589.7	2148.3	Positive*	Positive*
Apply Security Requirements	28	558.1	2055.5	Negative	Positive
Improve Development Process	28	519.5	2121.5	Negative	Negative
Perform Security Testing	28	192.2	456.5	Negative	Positive*
Apply Security Tooling	29	184.6	429.4	Negative*	Positive*
Apply Secure Coding Standards	29	168.4	326.2	Negative*	Positive*
Track Vulnerabilities	29	152.7	204.8	Negative*	Positive*
Perform Security Review	30	122.3	167.2	Negative*	Positive*
Apply Data Classification Scheme	27	55.0	148.9	Positive*	Positive*
Perform Security Training	28	32.1	73.6	Positive	Negative
Publish Operations Guide	25	21.9	48.8	Positive	Positive*

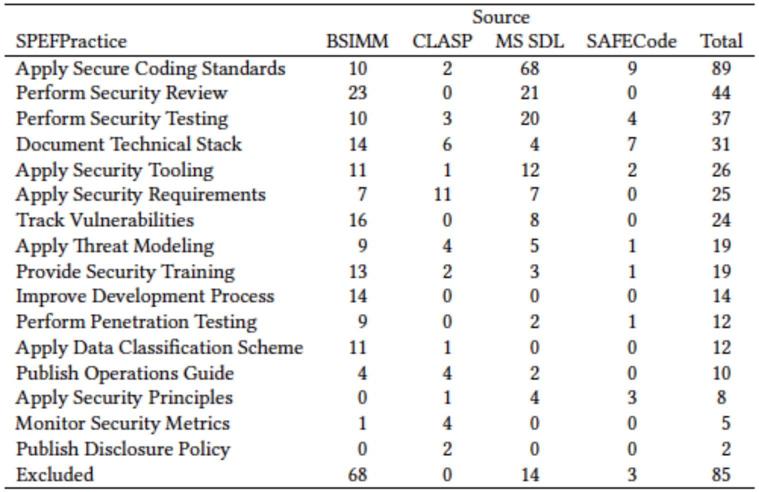
Limitations

- Our source lists of security practices are biased toward large organizations
- Our practice list and vocabulary was developed by a very small group, may need refinement
- Very low survey response rate (< 2%)
  - Our practice definitions may be unfamiliar
  - Our survey questions and instrument may need refinement
  - 'Your email states: "the survey is anonymous and your responses cannot be associated with you" but then survey ask for project names and github URLs. This would clearly deanonymize me.'

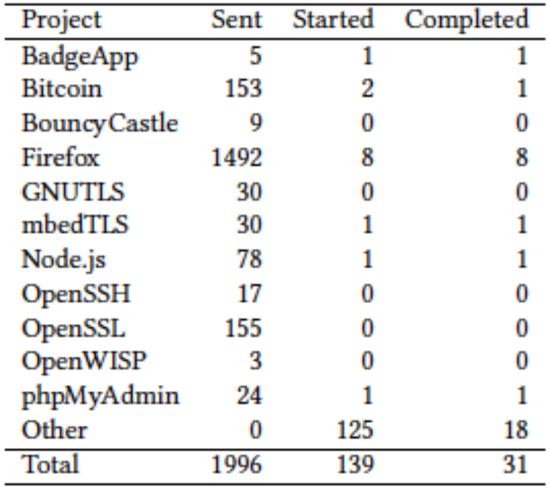
Summary

- Developed a list of core software development security practices
- Surveyed 11 software development teams, found evidence for use of these software development security practices
- Developed a set of practice adherence metrics
- In our data, training correlated with increased practice use

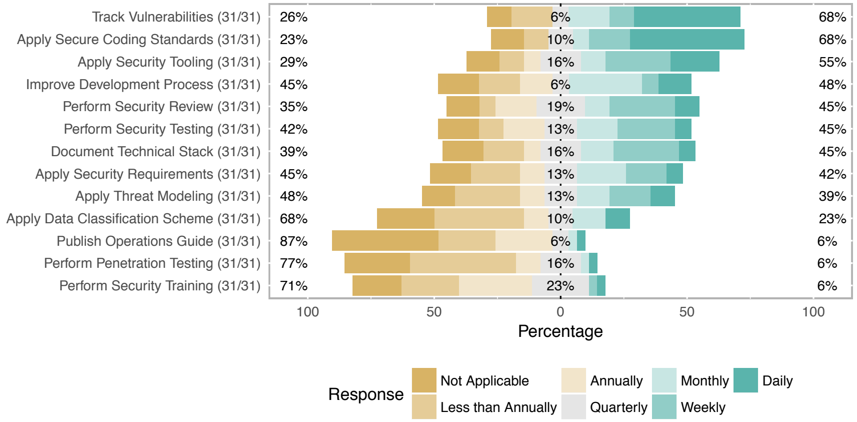
Summary



	Source				
SPEFPractice	BSIMM	CLASP	MS SDL	SAFECode	Total
Apply Secure Coding Standards	10	2	68	9	89
Perform Security Review	23	0	21	0	44
Perform Security Testing	10	3	20	4	37
Document Technical Stack	14	6	4	7	31
Apply Security Tooling	11	1	12	2	26
Apply Security Requirements	7	11	7	0	25
Track Vulnerabilities	16	0	8	0	24
Apply Threat Modeling	9	4	5	1	19
Provide Security Training	13	2	3	1	19
Improve Development Process	14	0	0	0	14
Perform Penetration Testing	9	0	2	1	12
Apply Data Classification Scheme	11	1	0	0	12
Publish Operations Guide	4	4	2	0	10
Apply Security Principles	0	1	4	3	8
Monitor Security Metrics	1	4	0	0	5
Publish Disclosure Policy	0	2	0	0	2
Excluded	68	0	14	3	85



Project	Sent	Started	Completed
BadgeApp	5	1	1
Bitcoin	153	2	1
BouncyCastle	9	0	0
Firefox	1492	8	8
GNUTLS	30	0	0
mbedTLS	30	1	1
Node.js	78	1	1
OpenSSH	17	0	0
OpenSSL	155	0	0
OpenWISP	3	0	0
phpMyAdmin	24	1	1
Other	0	125	18
Total	1996	139	31



pjmorris@ncsu.edu, @pmorrison,