Securing Agile: Assessing the Impact of Security on Agile Development

•••

Ryan Sajulga
sajul004@morris.umn.edu
16 November 2024

Motivation

Motivation

- Learn more about security
- Career in cybersecurity
- Not emphasized in the CSci Curriculum

Outline

- Background
  - Agile Methodology
  - Security
- Methodology
  - Survey
- Results
- Discussion
- Final Thoughts

Main Source

Year: 2024

By Arpit Thool & Chris Brown

Background

What do you do when you have too much to do, and not enough time?

Make a list!

You make a list

To Do List

- ❏ Clean houses
- ❏ Do Dishes
- ❏ Get groceries
- ❏ Journal
- ❏ Workout
- ❏ Carwash
- ❏ Read a book

Size Things up

To Do List

- ❏ Clean house (1 hr)
- ❏ Do Dishes (.5 hr)
- ❏ Get groceries (1 hr)
- ❏ Journal (1 hr)
- ❏ Workout (1.5 hr)
- ❏ Carwash (1 hr)
- ❏ Read a book (.5 hr)

Set Priorities

To Do List (Most Important)

- ❑ Clean house (1 hr)
- ❑ Do Dishes (.5 hr)
- ❑ Get groceries (1 hr)
- ❑ Workout (1.5 hr)

—--

(Least Important)

- ❑ Carwash (1 hr)
- ❑ Read a book (.5 hr)
- ❑ Journal (1 hr)

Execute

To Do List

- ✓ Clean house (1 hr)
- ✓ Do Dishes (.5 hr)
- ✓ Get groceries (1 hr)
- ❑ Workout (1.5 hr)

—--

- ❑ Carwash (1 hr)
- ❑ Read a book (.5 hr)
- ❑ Journal (1 hr)

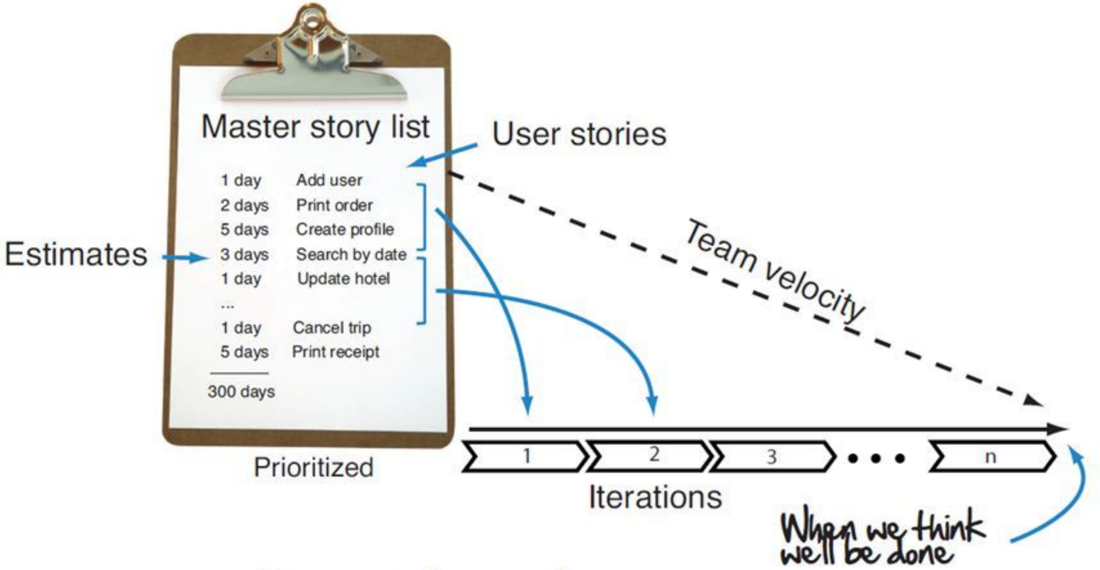
Basically the same idea in Agile methodology

To Do List → Master Story Lists

Tasks → User Stories

Guesses → Estimates

Agile Planning



Master story list

1 day	Add user
2 days	Print order
5 days	Create profile
3 days	Search by date
1 day	Update hotel
...	
1 day	Cancel trip
5 days	Print receipt
300 days	

User stories

Estimates

Prioritized

Team velocity

Iterations

When we think we'll be done

Agile principle

Working software is the primary measure of success.

https://slideplayer.com/slide/3924035/

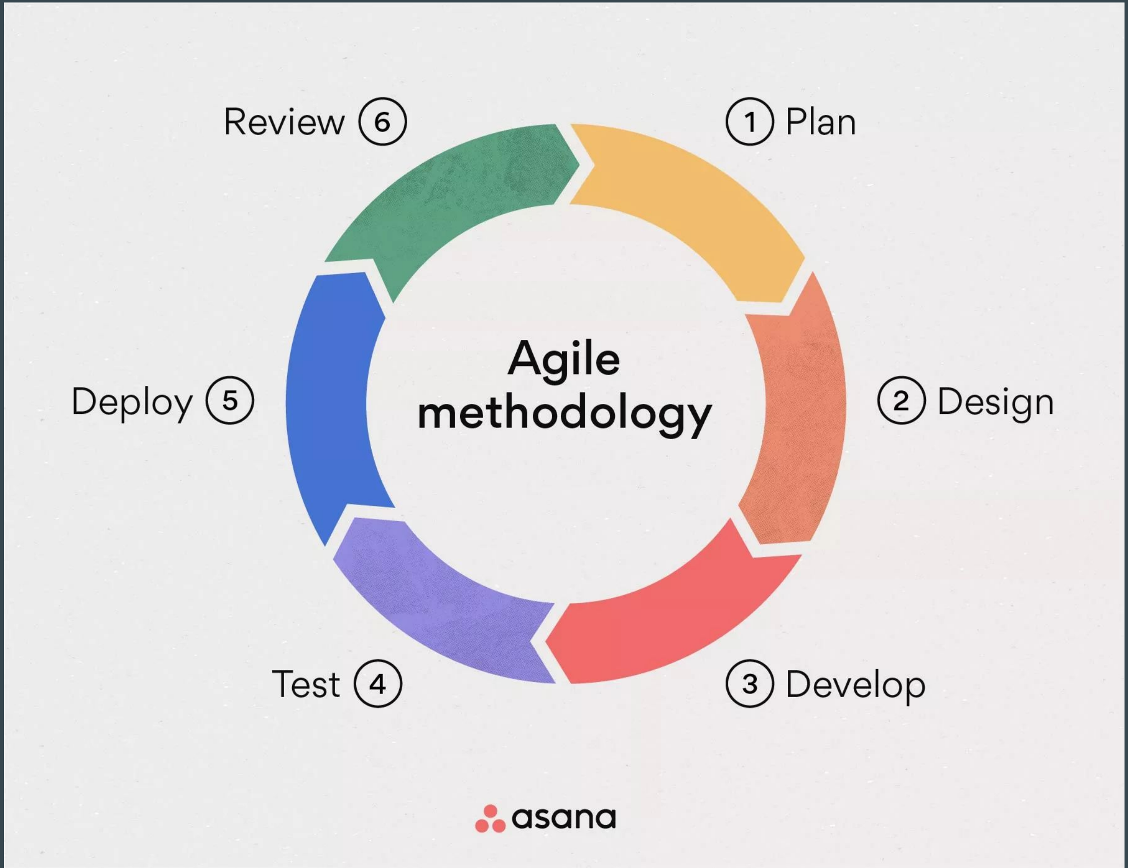
What is Agile Methodology?

- Project management methodology
- Main Focuses:
  - Flexibility
  - Collaboration
  - Rapid iteration
- Based off Agile Manifesto

Agile Manifesto - Core Values

We value...

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation.
3. Customer Collaboration over contract negotiation
4. Responding to change over following a plan



https://asana.com/resources/agile-methodology

Cybersecurity

Definition

Cyber Security

An umbrella term for the methods and strategies used to protect hardware, software, data and other internet-connected systems

Why is Security so Important?

- Keeps users’ data protected
- Prevents cyber attacks
- Maintains system integrity
- Compromised software can lead to consequences
    - Data leaks

The Equifax Data Breach

- May 2017
- One of largest credit reporting agencies
- Personally identified information (PII) leaked (~147 million users)
- Billions of dollars lost in the market value
- Sparked an interest in security

https://www.facebook.com/Equifax/

Eight Security Activities

Security Activity #1

“Addressing security in early iterations with requirements and testing”

Definition

This security activity emphasizes the importance of development teams addressing security issues and concerns early in the project before deploying the software

Security Activity #2

“Stating security requirements that are expected in the production software”

Definition

This requires incorporating security expectations in project requirements when describing the responsibilities and behavior of the software.

Security Activity #3

“Adding a security specialist to your team”

Definition

Security specialists, such as a Security Master, are members of a development team that focus on security aspects of the project to address concerns and ensure the security of the system

Security Activity #4

“Additional points or weights to issues with an impact on security”

Definition

This activity involves increasing the weights, such as story points in an Agile development environment, of issues that will have a higher impact the security of the product to prioritize security-related tasks and encourage more secure development and testing

Security Activity #5

“Iterative and incremental vulnerability and penetration testing”

Definition

This security activity suggests incorporating recurring security scanning, such as Dynamic Application Security Testing (DAST), to test for security flaws in the working software automatically

Security Activity #6

“Iterative and incremental security static analysis”

Definition

Similar to DAST, Static Application Security Testing (SAST) involves using security-related static analysis tools to detect potential security vulnerabilities by scanning the source code

Security Activity #7

“Iterative and incremental risk analysis, countermeasure graphs”

Definition

This security activity consists of using tools to monitor networks, applications, and infrastructure and perform risk analysis to identify vulnerabilities. These tools can evaluate the system’s security and suggest methods to prevent attacks

Security Activity #8

“Automatic testing”

Definition

This security activity involves incorporating secure coding practices, such as vulnerabilities analysis and risk assessment, into the deployment pipeline for software projects. This allows security checks to be automatically triggered with code changes and issues to be addressed before the software is deployed to users.

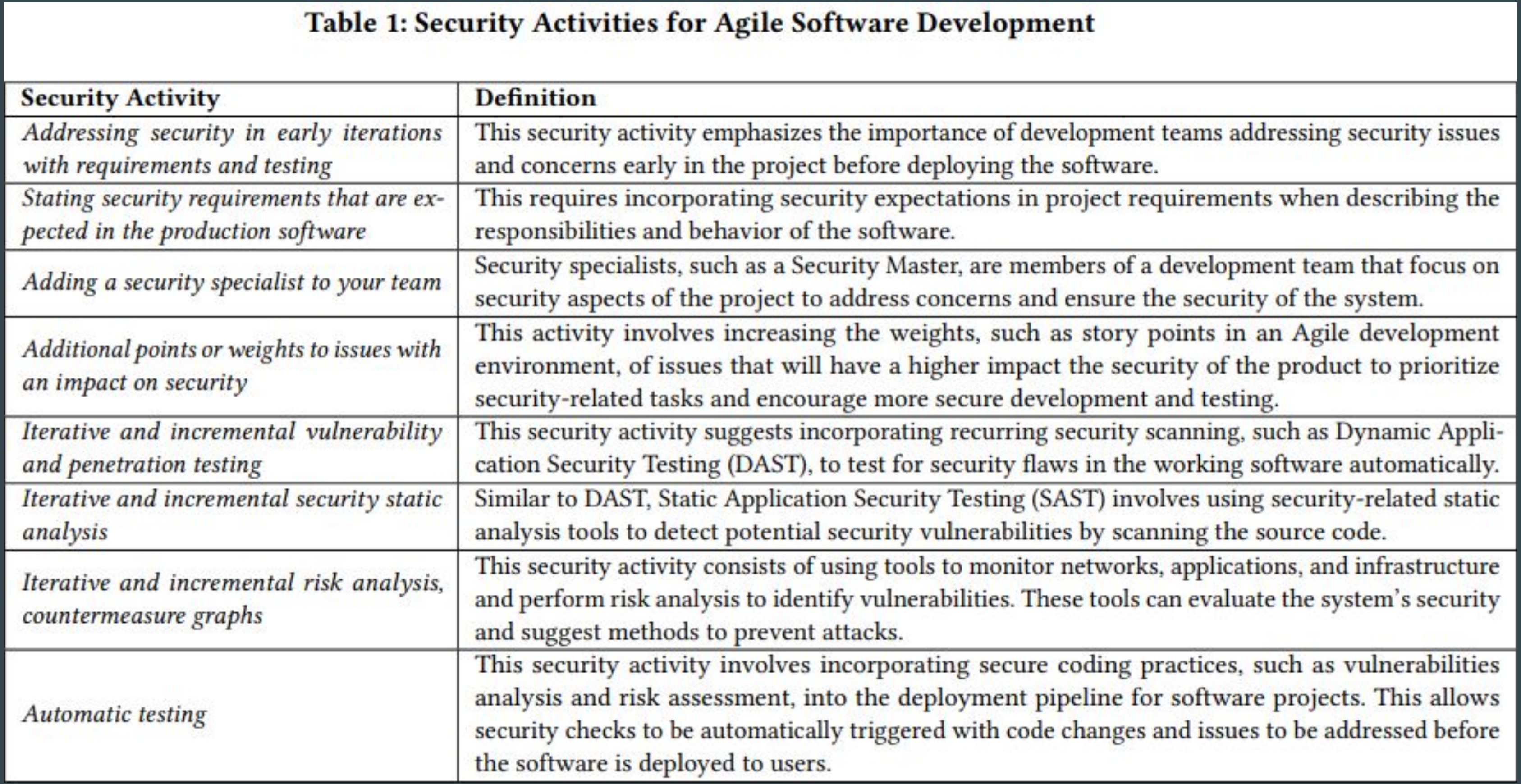


Table 1: Security Activities for Agile Software Development

Security Activity	Definition
Addressing security in early iterations with requirements and testing	This security activity emphasizes the importance of development teams addressing security issues and concerns early in the project before deploying the software.
Stating security requirements that are expected in the production software	This requires incorporating security expectations in project requirements when describing the responsibilities and behavior of the software.
Adding a security specialist to your team	Security specialists, such as a Security Master, are members of a development team that focus on security aspects of the project to address concerns and ensure the security of the system.
Additional points or weights to issues with an impact on security	This activity involves increasing the weights, such as story points in an Agile development environment, of issues that will have a higher impact the security of the product to prioritize security-related tasks and encourage more secure development and testing.
Iterative and incremental vulnerability and penetration testing	This security activity suggests incorporating recurring security scanning, such as Dynamic Application Security Testing (DAST), to test for security flaws in the working software automatically.
Iterative and incremental security static analysis	Similar to DAST, Static Application Security Testing (SAST) involves using security-related static analysis tools to detect potential security vulnerabilities by scanning the source code.
Iterative and incremental risk analysis, countermeasure graphs	This security activity consists of using tools to monitor networks, applications, and infrastructure and perform risk analysis to identify vulnerabilities. These tools can evaluate the system's security and suggest methods to prevent attacks.
Automatic testing	This security activity involves incorporating secure coding practices, such as vulnerabilities analysis and risk assessment, into the deployment pipeline for software projects. This allows security checks to be automatically triggered with code changes and issues to be addressed before the software is deployed to users.

Methodology

Data Collection

- Online Survey
- Created in QuestionPro
- Nine questions
- Gather information

https://www.questionpro.com/us/?

Participation Recruitment

- Experience with Agile development
- Reach out methods
  - Personalized invites
  - Posts on LinkedIn
  - Slack
- 34 Participants Total
  - 67% (23) had average of 8 years of technical work experience
  - 33% (11) were university students pursuing graduate studies

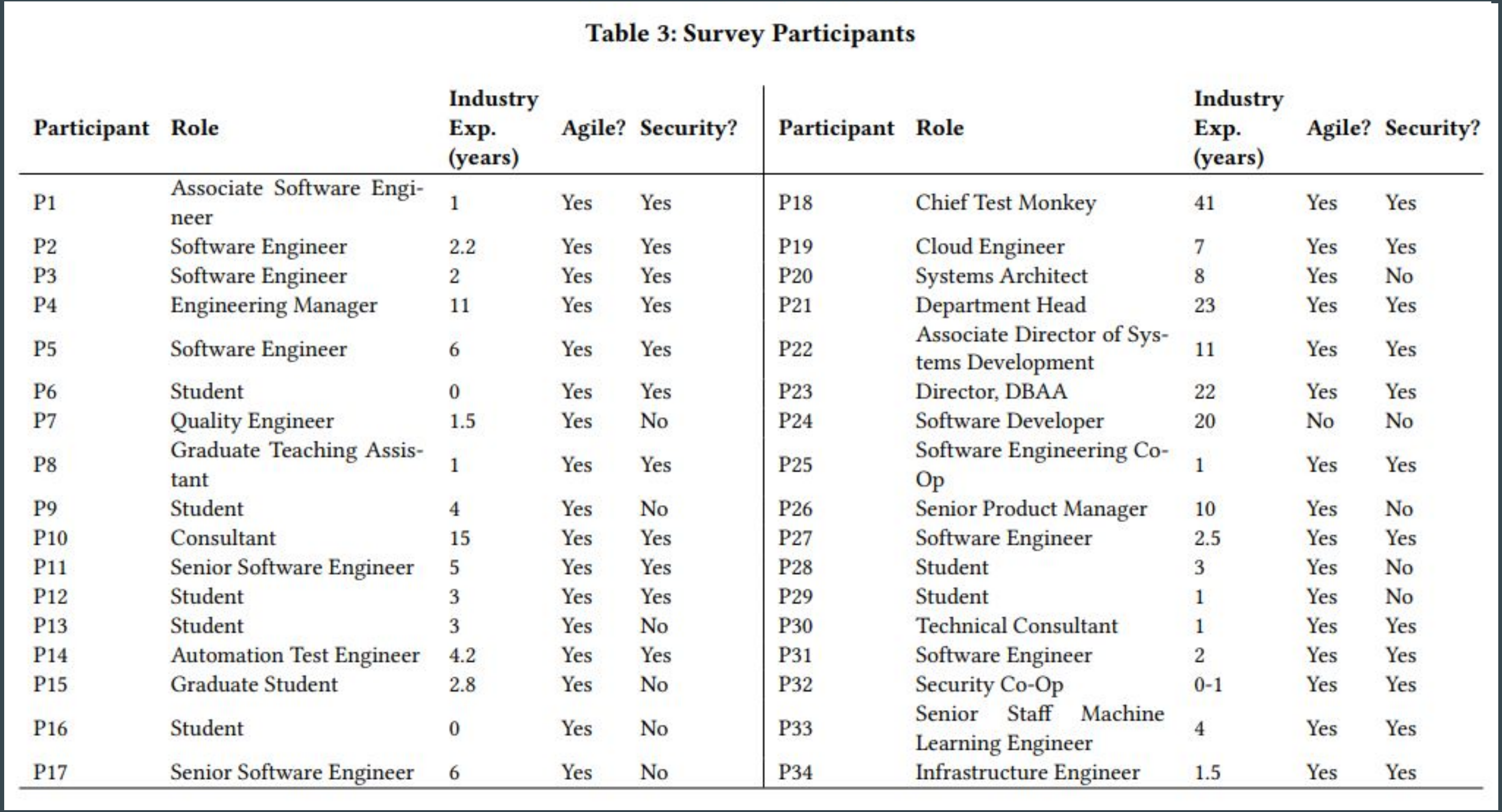


Table 3: Survey Participants

Participant	Role	Industry Exp. (years)	Agile?	Security?
P1	Associate Software Engineer	1	Yes	Yes
P2	Software Engineer	2.2	Yes	Yes
P3	Software Engineer	2	Yes	Yes
P4	Engineering Manager	11	Yes	Yes
P5	Software Engineer	6	Yes	Yes
P6	Student	0	Yes	Yes
P7	Quality Engineer	1.5	Yes	No
P8	Graduate Teaching Assistant	1	Yes	Yes
P9	Student	4	Yes	No
P10	Consultant	15	Yes	Yes
P11	Senior Software Engineer	5	Yes	Yes
P12	Student	3	Yes	Yes
P13	Student	3	Yes	No
P14	Automation Test Engineer	4.2	Yes	Yes
P15	Graduate Student	2.8	Yes	No
P16	Student	0	Yes	No
P17	Senior Software Engineer	6	Yes	No
P18	Chief Test Monkey	41	Yes	Yes
P19	Cloud Engineer	7	Yes	Yes
P20	Systems Architect	8	Yes	No
P21	Department Head	23	Yes	Yes
P22	Associate Director of Systems Development	11	Yes	Yes
P23	Director, DBAA	22	Yes	Yes
P24	Software Developer	20	No	No
P25	Software Engineering Co-Op	1	Yes	Yes
P26	Senior Product Manager	10	Yes	No
P27	Software Engineer	2.5	Yes	Yes
P28	Student	3	Yes	No
P29	Student	1	Yes	No
P30	Technical Consultant	1	Yes	Yes
P31	Software Engineer	2	Yes	Yes
P32	Security Co-Op	0-1	Yes	Yes
P33	Senior Staff Machine Learning Engineer	4	Yes	Yes
P34	Infrastructure Engineer	1.5	Yes	Yes

Survey Structure

- Designed to collect background info
- Experiences with Agile development
- Goal to answer three research questions

- The Survey contains...
- Closed-ended
- Likert Scale
- Open-ended

Research Questions

Research Question 1 (pt 1)

How do software practitioners perceive the effectiveness of adopted and state-of-the-art security practices.

Research Question 1 (pt 2)

What is their level of willingness to incorporate them into the Agile software development process?

Research Question 2

How are the team velocity and productivity, as perceived by the software practitioners, affected by the inclusion of security activities?

Research Question 3

What is the impact of integrating security activities into Agile development on software practitioners' confidence in their software product and organization?

Results

Survey Question - Background

Do you use agile software development methodology in your organization?

❑ Yes

❑ No

Results

- 97% (33) used Agile methodology

Survey Question - Background

Does your team include any security activities in the Agile process?

❏ Yes

❏ No

Results

- 72% (23) reported having security-related practices in their process

Survey Question - RQ1

What is your take on these security practices used in your team? (Optional)

Results

- Good (8)
- Informative (2)
- Necessary (2)
- Needs to be Complied with (1)
- Time-consuming (1)
- Disliked (1)
- Need for Improvement (4)

Survey Question - RQ1

How effective would each security practice be in increasing the security and robustness of the software, if your team would include it in the agile software development process?

(8 Security Practices)

❏ Not at all effective

❏ Slightly effective

❏ Moderately effective

❏ Very effective

❏ Extremely effective

Perceived Effectiveness

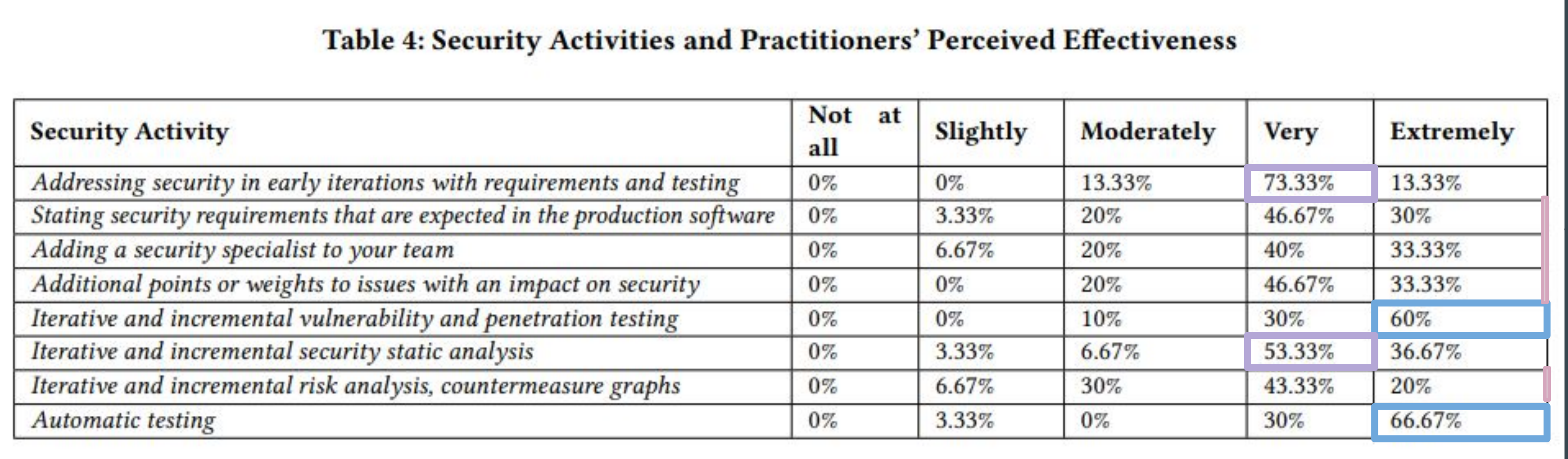


Table 4: Security Activities and Practitioners' Perceived Effectiveness

Security Activity	Not at all	Slightly	Moderately	Very	Extremely
Addressing security in early iterations with requirements and testing	0%	0%	13.33%	73.33%	13.33%
Stating security requirements that are expected in the production software	0%	3.33%	20%	46.67%	30%
Adding a security specialist to your team	0%	6.67%	20%	40%	33.33%
Additional points or weights to issues with an impact on security	0%	0%	20%	46.67%	33.33%
Iterative and incremental vulnerability and penetration testing	0%	0%	10%	30%	60%
Iterative and incremental security static analysis	0%	3.33%	6.67%	53.33%	36.67%
Iterative and incremental risk analysis, countermeasure graphs	0%	6.67%	30%	43.33%	20%
Automatic testing	0%	3.33%	0%	30%	66.67%

Survey Question - RQ1

How willing are you to include each security practice in your Agile software development process?

(8 Security Practices)

- ❏ Not at all willing
- ❏ Slightly willing
- ❏ Moderately willing
- ❏ Very willing
- ❏ Extremely willing

Willingness to Include

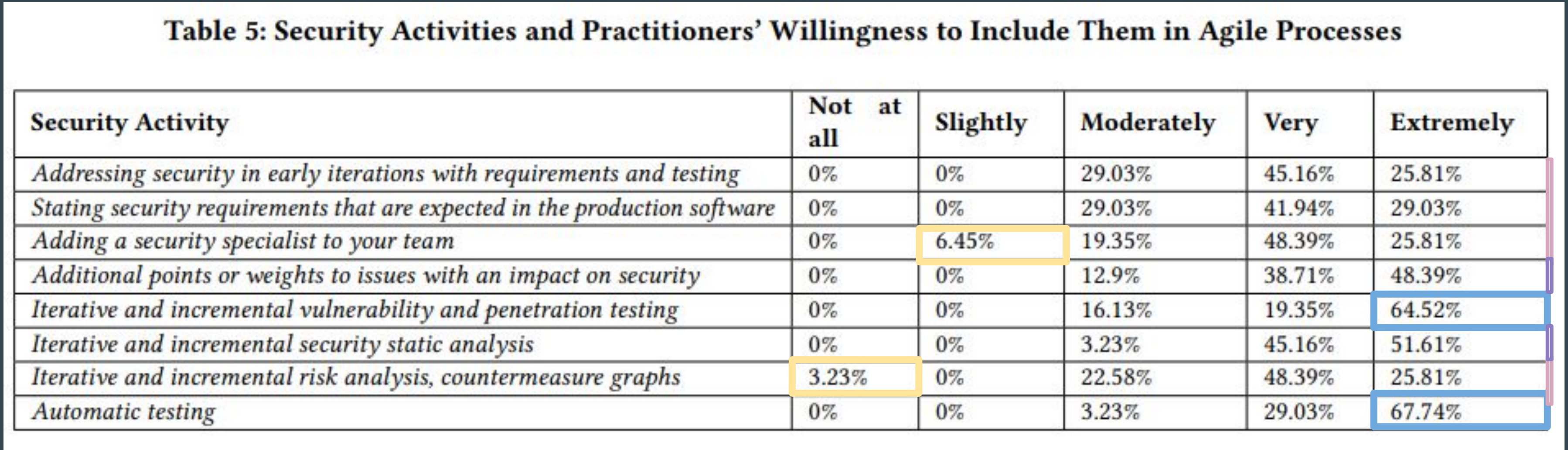


Table 5: Security Activities and Practitioners' Willingness to Include Them in Agile Processes

Security Activity	Not at all	Slightly	Moderately	Very	Extremely
Addressing security in early iterations with requirements and testing	0%	0%	29.03%	45.16%	25.81%
Stating security requirements that are expected in the production software	0%	0%	29.03%	41.94%	29.03%
Adding a security specialist to your team	0%	6.45%	19.35%	48.39%	25.81%
Additional points or weights to issues with an impact on security	0%	0%	12.9%	38.71%	48.39%
Iterative and incremental vulnerability and penetration testing	0%	0%	16.13%	19.35%	64.52%
Iterative and incremental security static analysis	0%	0%	3.23%	45.16%	51.61%
Iterative and incremental risk analysis, countermeasure graphs	3.23%	0%	22.58%	48.39%	25.81%
Automatic testing	0%	0%	3.23%	29.03%	67.74%

Survey Question - RQ2

How was the sprint velocity affected? (Optional)

Team Velocity

- About 14 people responded
- Overall, adopting security practices did not affect the teams output

Survey Question - RQ2

How has the involvement of these security practices affected your day-to-day activities? (Optional)

Day-to-day Activities

- Less effect on their daily work
- No major impact on sprint velocity

Survey Question - RQ3

How has the involvement of these security practices affected the software product? (Optional)

Software Products

- About 10 participants reported
- Involving security practices increases the overall security of products
- Negative Effects
    - (P30) “Extended delivery date since code was often stuck waiting for approval”
    - (P19) “Adopting security practices rarely impacted the products’s security”

Survey Question - RQ3

How has the inclusion of these security practices affected the organization? (Optional)

Organizations

Few Positive Effects

- Improved overall culture (1)
- Build company reputation (1)
- Increased customer confidence (1)

Neutral

- No effect (4)
- Minimal impact (4)

Survey Question - RQ3

After using these security practices are you more confident in the security of the software you are building?

Confidence

❏ Not confident at all

❏ Slightly confident

❏ Somewhat confident

❏ Fairly confident

❏ Completely confident

RQ3 Continued...

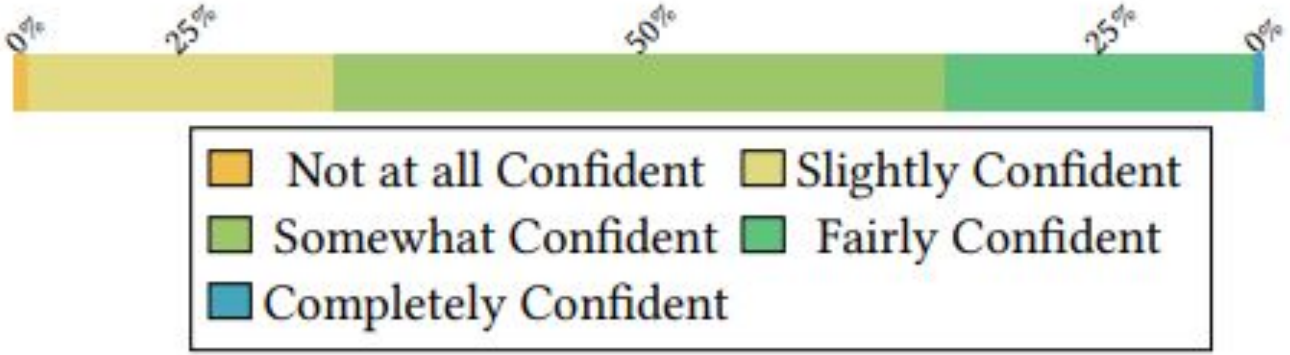


Figure 2: Participants' confidence in software security was influenced by the adoption of security practices by their Agile teams

Discussion

Summary

Positives

- Positively perceived despite the potential of conflicts
- Growing awareness of the importance of software security
- Integration has minimal impact on productivity
- Increase security of software products

Negatives

- Increased time
- Occasional delays
- Confidence for some participants were “somewhat” or “fairly”

Two Suggestions...

Increase Automation

- Highlighted with survey
- Preferred over manual
    - Security Specialist
- Can be harmful
    - Inaccurate output
    - Not meeting stakeholder requirements
    - Information overload

Improved Feedback

- Blind automation is not effective
- Clear, actionable feedback help with confidence
- Guidance on fixing vulnerabilities

Final Thoughts

- Security practices is useful
- Did not negativity impact productivity
- Needs more research, but provides insight
- Limitations
  - Number of participants
  - Students from same school (Virginia Tech)
  - Relies on memory and estimations

Questions?