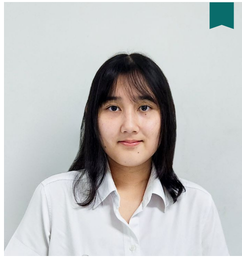
ON CATEGORIZING OPEN SOURCE SOFTWARE SECURITY VULNERABILITY REPORTING MECHANISMS ON GITHUB

Scan QR code to
see our paper

Sushawapak
Kancharoendee



Thanat
Phichitphanphong

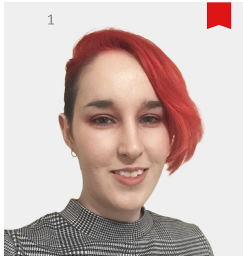
Chanikarn
Jongyingyos

Raula
Gaikovina Kula

Morakot
Choetkiertikul

Chaiyong
Ragkhitwetsagul

Thanwadee
Sunetnanta



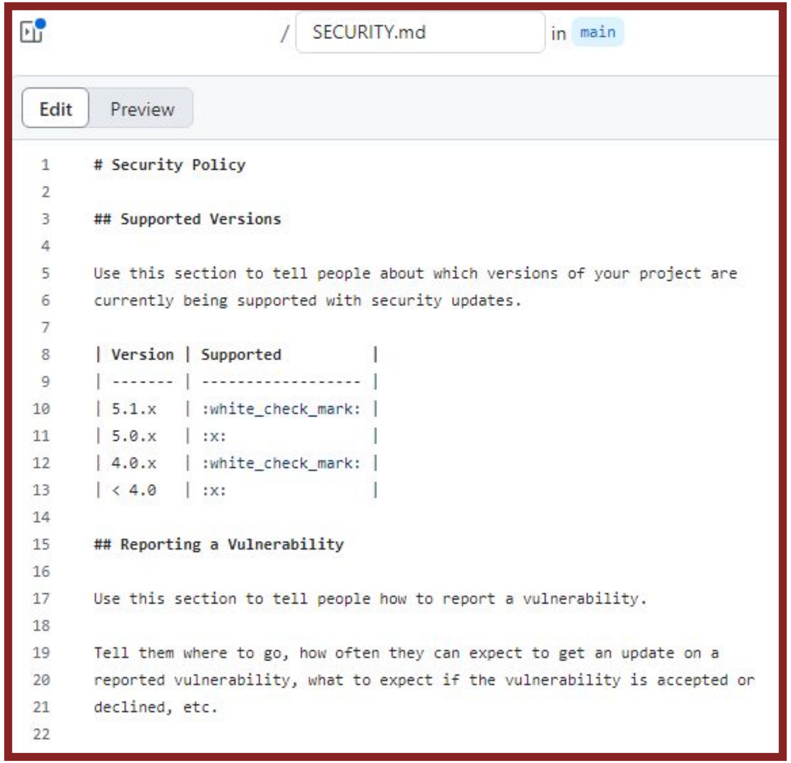
Brittany
Reid

Mahidol
University

大阪大学
OSAKA UNIVERSITY

国立大学法人
奈良先端科学技術大学院大学
NARA INSTITUTE of SCIENCE and TECHNOLOGY

GitHub projects often contain security policies:



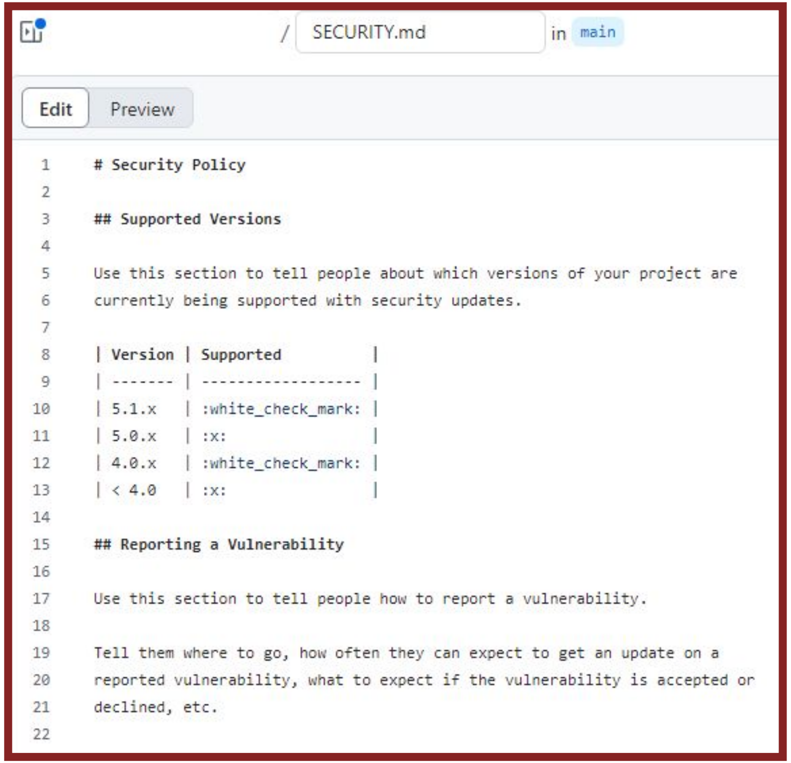
```
# Security Policy

## Supported Versions

Use this section to tell people about which versions of your project are
currently being supported with security updates.

| Version | Supported          |
| ------- | ------------------ |
| 5.1.x   | :white_check_mark: |
| 5.0.x   | :x:                |
| 4.0.x   | :white_check_mark: |
| < 4.0   | :x:                |

## Reporting a Vulnerability

Use this section to tell people how to report a vulnerability.

Tell them where to go, how often they can expect to get an update on a
reported vulnerability, what to expect if the vulnerability is accepted or
declined, etc.

```

GitHub security policy template

Provide instructions for **reporting security vulnerabilities** in the project

Variety of mechanisms such as **email**, **GitHub advisories** and **external platforms**

GitHub projects often contain security policies:

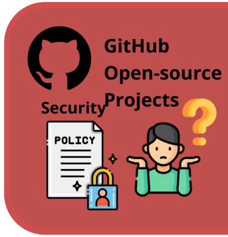


```
# Security Policy

## Supported Versions

Use this section to tell people about which versions of your project are
currently being supported with security updates.

| Version | Supported          |
| ------- | ------------------ |
| 5.1.x   | :white_check_mark: |
| 5.0.x   | :x:                |
| 4.0.x   | :white_check_mark: |
| < 4.0   | :x:                |

## Reporting a Vulnerability

Use this section to tell people how to report a vulnerability.

Tell them where to go, how often they can expect to get an update on a
reported vulnerability, what to expect if the vulnerability is accepted or
declined, etc.

```

GitHub security policy template

Provide instructions for **reporting security vulnerabilities** in the project

Variety of mechanisms such as **email**, **GitHub advisories** and **external platforms**



There are currently no studies examining the specific **characteristics of security policies** in open-source projects.

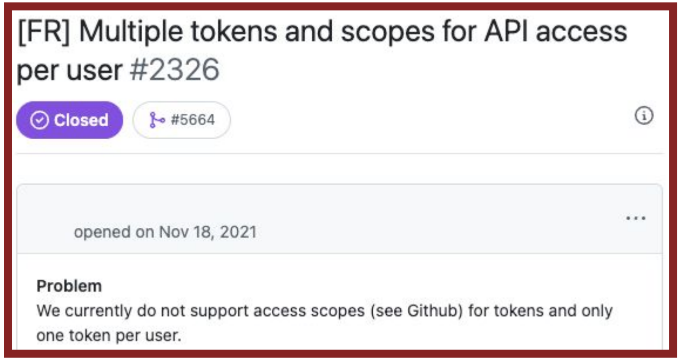
Why do we want to know about security policies?

- We want to **understand the commonly recommended security reporting mechanisms** on GitHub
- We want to know **if developers follow these mechanisms or not**
- And we want to know if projects with security policies **are more secure**

Why do we want to know about security policies?

- We want to **understand the commonly recommended security reporting mechanisms** on GitHub
- We want to know **if developers follow these mechanisms or not**
- And we want to know if security policies **make projects more secure**

Not all security reporting mechanisms are good...



[FR] Multiple tokens and scopes for API access per user #2326

opened on Nov 18, 2021

Problem
We currently do not support access scopes (see Github) for tokens and only one token per user.

Insecure reporting mechanisms can expose vulnerabilities to attackers

RQ1: What are the reporting mechanisms in security policies?

We look at **679 PyPI packages** with that appear in the GitHub advisory database.

For the **303 (44.6%) with a security policy**:

- We manually classify the reporting mechanism:

RQ1: What are the reporting mechanisms in security policies?

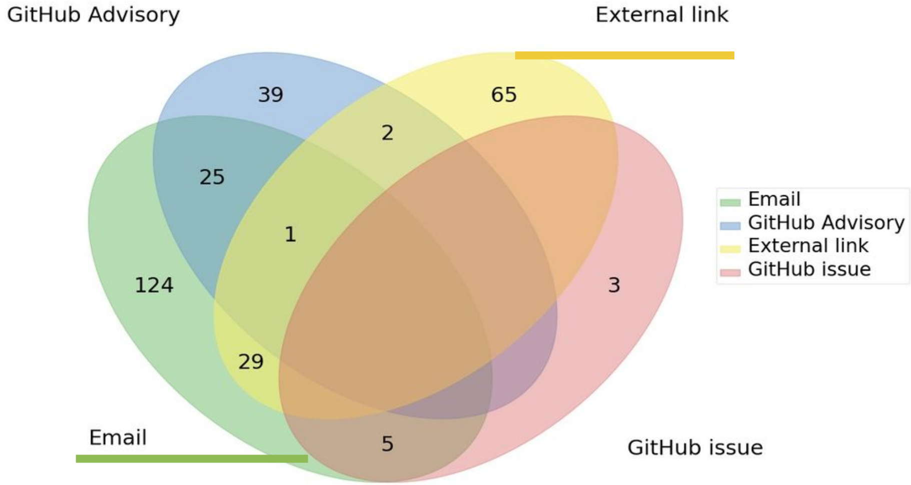
We look at **679 PyPI packages** with that appear in the GitHub advisory database.

For the **303 (44.6%) with a security policy**:

- We manually classify the reporting mechanism:

We find that:

- Most repositories use **Email (41.06%)**, or **External links (21.52%)**



Venn diagram of the reporting mechanisms defined in security policies

RQ1: What are the reporting mechanisms in security policies?

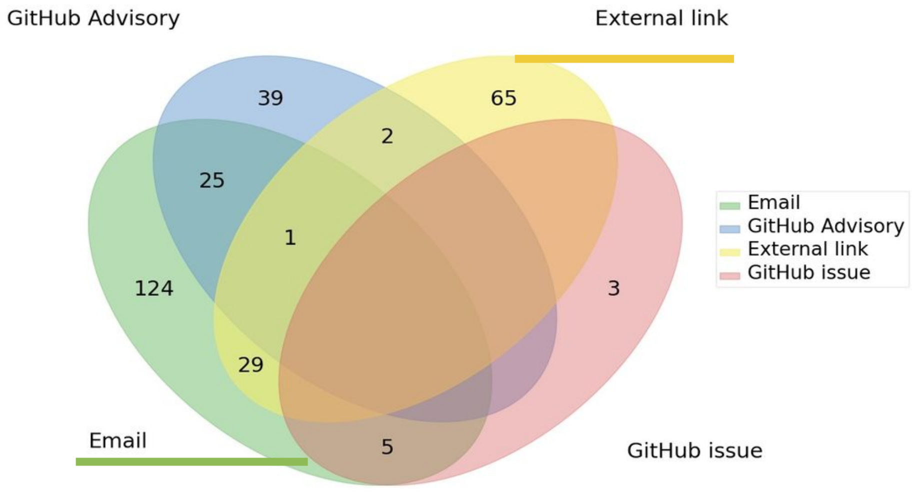
We look at **679 PyPI packages** with that appear in the GitHub advisory database:

For the **303 (44.6%) with a security policy**:

- We manually classify the reporting mechanism:

We find that:

- Most repositories use **Email (41.06%)**, or **External links (21.52%)**



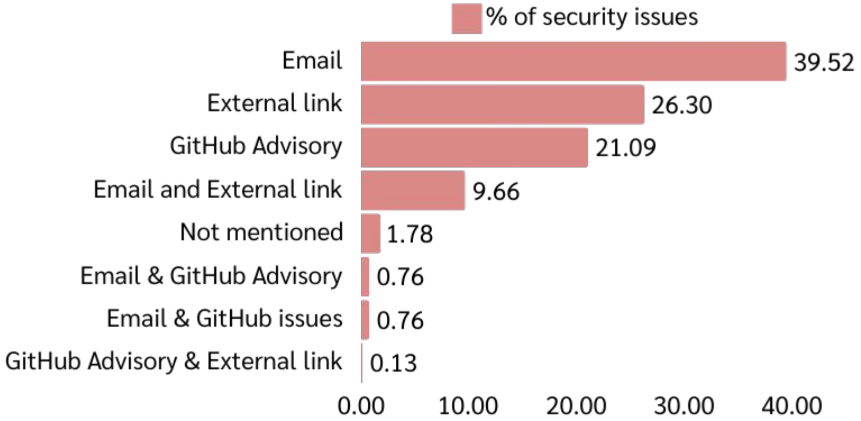
Venn diagram of the reporting mechanisms defined in security policies

Most projects maintainers are aware of the risk of publicly disclosed vulnerabilities, since most security policy reporting mechanisms are **private communication channels.**

RQ2: Do developers' practices align with the security policies?

We look for the existence of GitHub issues labeled *"vulnerability", "security", "risk", "CVE", "CWE"* etc. and find:

- **787 issues** non-compliant with security policies **across 58 repositories**

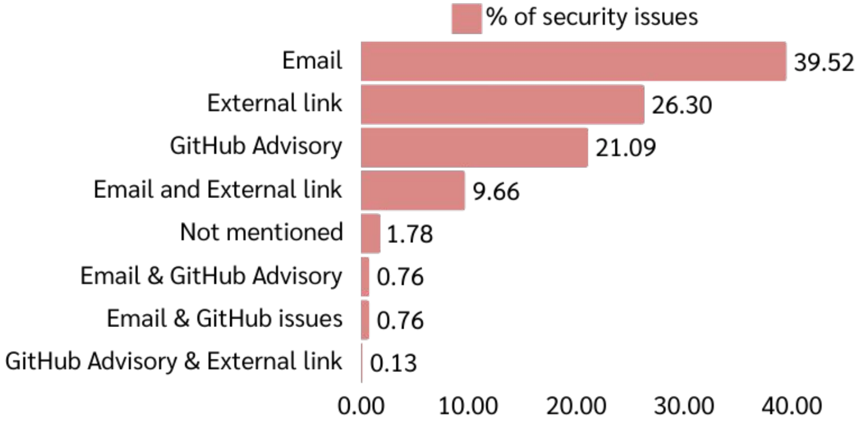


Distribution of non-compliant issues by mechanism defined in README

RQ2: Do developers' practices align with the security policies?

We look for the existence of GitHub issues labeled *"vulnerability", "security", "risk", "CVE", "CWE"* etc. and find:

- **787 issues** non-compliant with security policies **across 58 repositories**



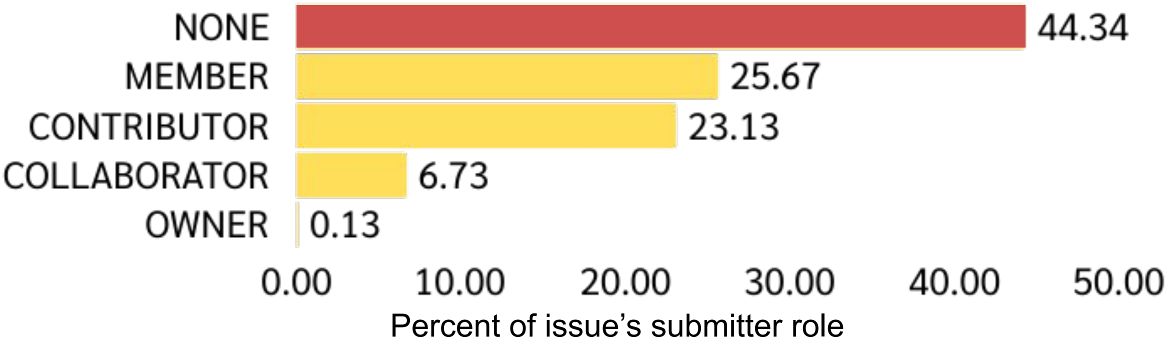
Distribution of non-compliant issues by mechanism defined in README

Developers may be reporting vulnerabilities with **insecure methods.**

RQ2: Do developers' practices align with the security policies?

However, when we look at the **role** of the issue submitter in the repository, we find:

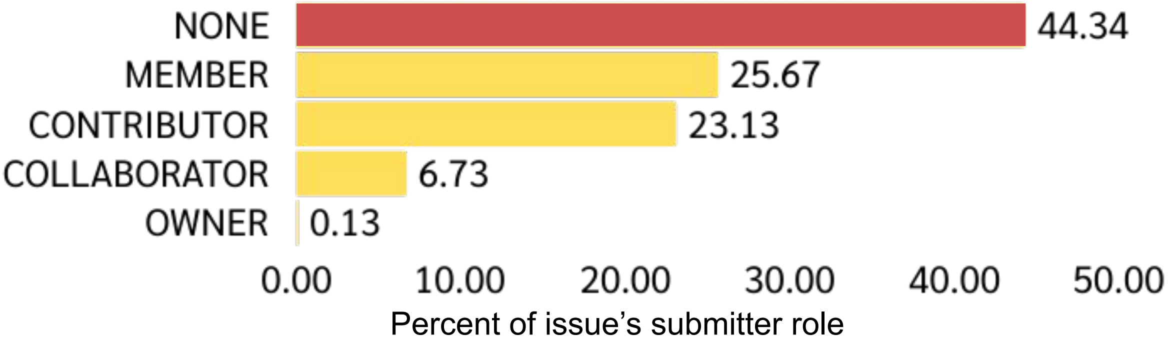
- **44.34%** have no **`author_association`** role



RQ2: Do developers' practices align with the security policies?

However, when we look at the **role** of the issue submitter in the repository, we find:

- **44.34%** have no `author_association` role

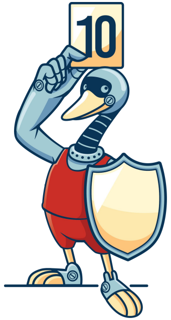


The most non-compliant issues are **created by external contributors**.

RQ3: Do projects with a security policy differ in OpenSSF Scorecard scores compared to those without one?

For **303 repositories with a security policy**, and **376 repositories without**:

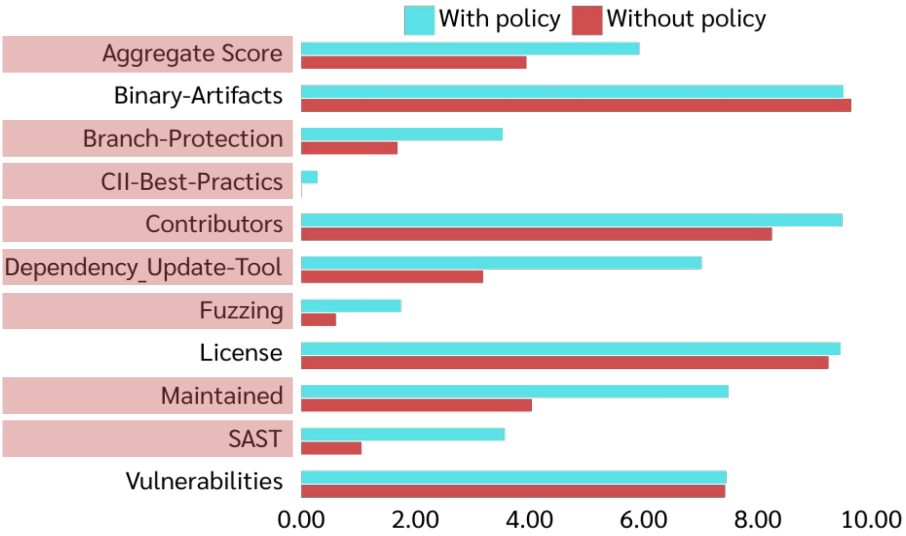
- We run the OpenSSF Scorecard tool and get 10 security criteria, and an aggregate score:



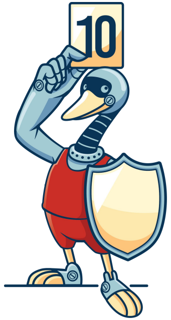
RQ3: Do projects with a security policy differ in OpenSSF Scorecard scores compared to those without one?

For 303 repositories with a security policy, and 376 repositories without:

- We run the OpenSSF Scorecard tool and get 10 security criteria, and an aggregate score:



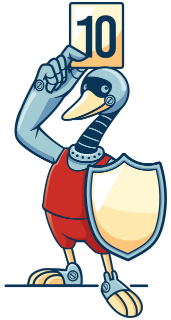
Average scores (out of 10) with and without policy. Criteria with a statistically significant difference are highlighted.

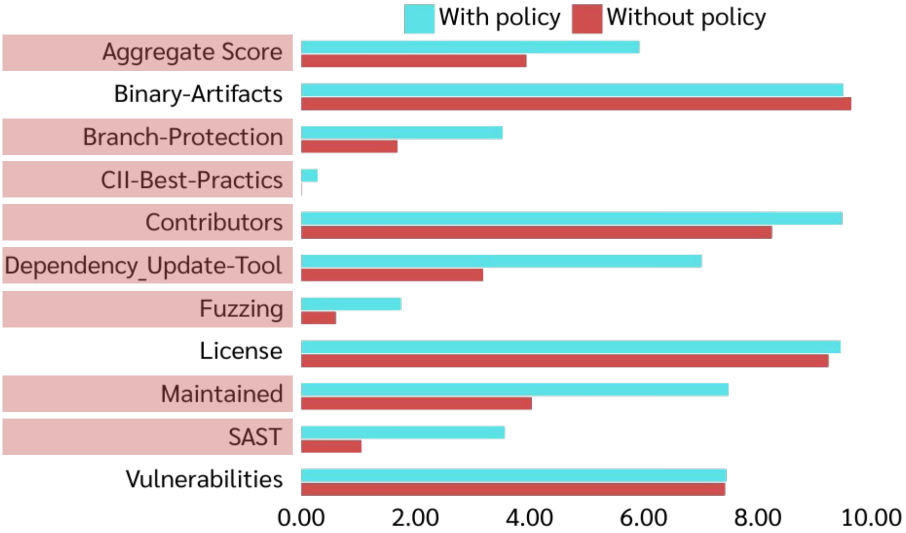


RQ3: Do projects with a security policy differ in OpenSSF Scorecard scores compared to those without one?

For **303 repositories with a security policy**, and **376 repositories without**:

- We run the OpenSSF Scorecard tool and get 10 security criteria, and an aggregate score:





Average scores (out of 10) with and without policy. Criteria with a statistically significant difference are highlighted.

Repositories with security policies are more proactive in implementing security practices.

CONCLUSION & FUTURE WORK

Most security policy reporting mechanisms use private communication channels, and projects with policies tends to adhere more closely to security practices in general.

Future directions...

Identify the best security policy practices across diverse ecosystems.

Explore automation and communication strategies for better adherence.