On the Benefits of Bug Bounty Programs: A Study of Chromium Vulnerabilities

Amutheezan Sivagnanam, Soodeh Atefi, Afiya Ayman, Jens Grossklags, and Aron Laszka

UNIVERSITY of HOUSTON

Technische Universität München

WEIS 2021

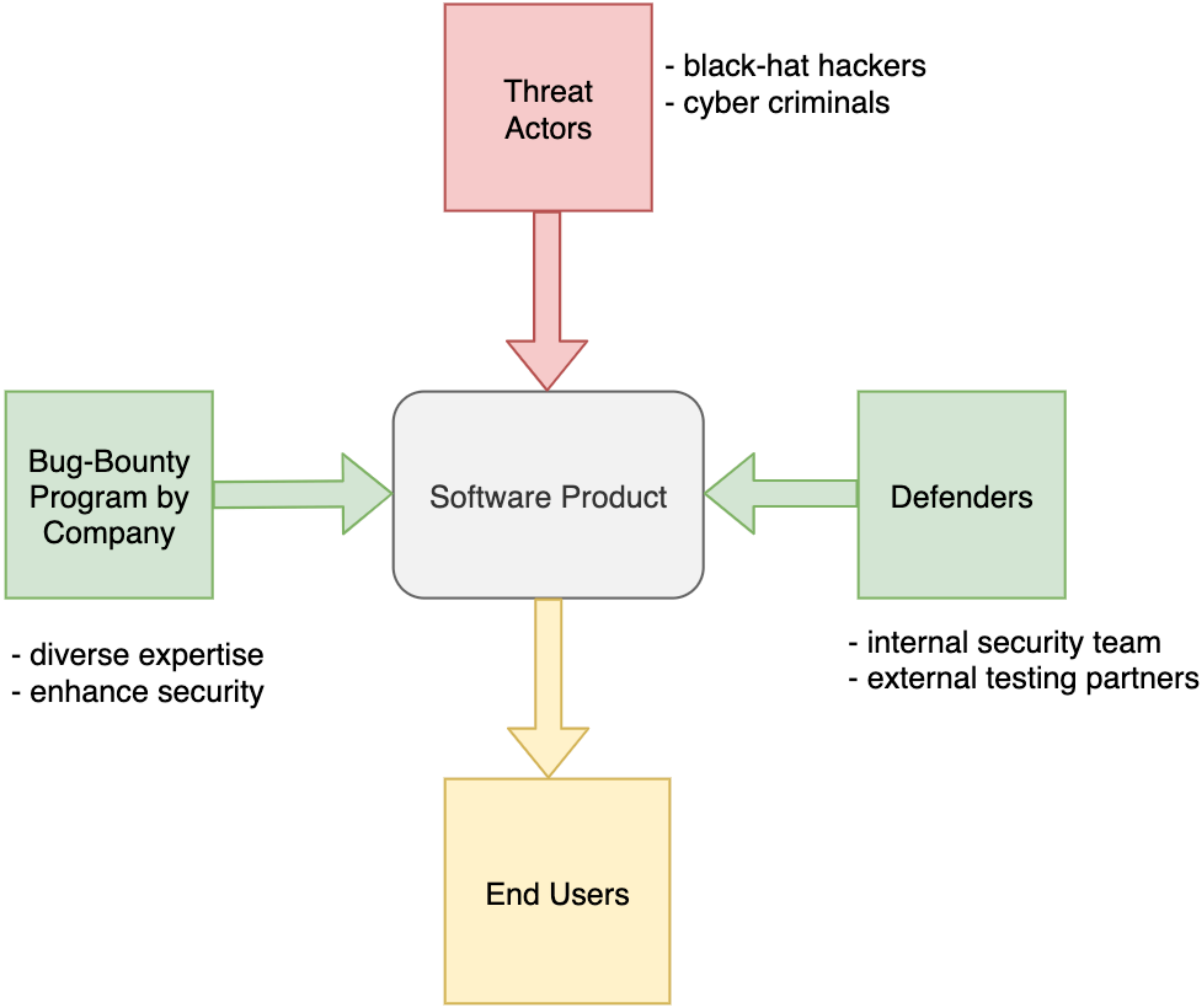
2021-06-28

Bug Bounty Programs

Software companies launch vulnerability reward programs (VRP) and allow external bug hunters with **diverse expertise** to test and report the vulnerabilities.

> e.g., Google, Intel, Facebook, and Microsoft

Based on the **validity/severity** of the report, the software company will reward the reporter.



Limitations in Previous Works

- Previous research efforts consider the value of bug bounty programs in terms of the number of reports made and technical aspects (e.g., severity).

  e.g., Finifter et al., Zhao et al., Maillart et al., Laszka et al., Luna et al., Elazari et al., Walshe and Simpson.

- But, the number of reported vulnerabilities and inherent properties of reports alone cannot quantify the security benefits of bug-bounty programs since they ignore the likelihood of discovery.

Our work looks into

what benefits bug hunters provide ? how does the probability of rediscovery vary based on different types of vulnerabilities ? are bug hunters finding a bug that would be exploited ?

Research Questions

Research Questions

RQ 1: Do external bug hunters **report vulnerabilities similar to those that internal testers report** ?

Research Questions

RQ 1: Do external bug hunters **report vulnerabilities similar to those that internal testers report** ?

Brady et al. suggest that there are efficiency benefits of testing software by different teams with diverse expertise. Votipka et al. report key differences between internal security testers and external bug hunters based on a survey from the reporter's perspective.

Research Questions

RQ 1: Do external bug hunters **report vulnerabilities similar to those that internal testers report** ?

Brady et al. suggest that there are efficiency benefits of testing software by different teams with diverse expertise. Votipka et al. report key differences between internal security testers and external bug hunters based on a survey from the reporter's perspective.

RQ 2: Does the probability of rediscovery is **negligible** ? How does rediscovery **vary between different types of vulnerabilities** ?

Research Questions

RQ 1: Do external bug hunters **report vulnerabilities similar to those that internal testers report** ?

Brady et al. suggest that there are efficiency benefits of testing software by different teams with diverse expertise. Votipka et al. report key differences between internal security testers and external bug hunters based on a survey from the reporter's perspective.

RQ 2: Does the probability of rediscovery is **negligible** ? How does rediscovery **vary between different types of vulnerabilities** ?

Rescorla et al. suggest that patching vulnerability can bring them to the threat actors' attention. But Schneier et al. claim that this only holds when the vulnerability rediscovery is negligible; otherwise, it needs to patch before threat actors discover it.

Research Questions

RQ 1: Do external bug hunters **report vulnerabilities similar to those that internal testers report** ?

Brady et al. suggest that there are efficiency benefits of testing software by different teams with diverse expertise. Votipka et al. report key differences between internal security testers and external bug hunters based on a survey from the reporter's perspective.

RQ 2: Does the probability of rediscovery is **negligible** ? How does rediscovery **vary between different types of vulnerabilities** ?

Rescorla et al. suggest that patching vulnerability can bring them to the threat actors' attention. But Schneier et al. claim that this only holds when the vulnerability rediscovery is negligible; otherwise, it needs to patch before threat actors discover it.

RQ 3: Do external bug hunters **report vulnerabilities similar to those that threat actors exploit** ?

We study Chromium, because

- It has a **long-running vulnerability reward program** (launched in January 2010)
- The source code is **open-source**, and the issue tracker is **publicly available**
- We build a comprehensive dataset from
  - **Chromium Issue Tracker** (vulnerability reports)
  - **CVE Details** (details of vulnerabilities, such as weakness type)
  - **Google Git** (details of files that changed to fix the vulnerabilities)
  - **Chrome Releases Blog** (list of issues patched in each release)
- We collect a total of **21,422** security reports from **September 2, 2008 to February 26, 2021**
  - **17,826** valid original security reports
  - **3,343** valid duplicate security reports

Data Cleaning

- **External vs. Internal Reports -** we identify the reporter origin using the reporter email (e.g., ends with @chromium.org, @google.com), comments, and chrome release notes

- **Manual vs. Automatic Reports -** we identify the automated reports (e.g., fuzzing tool) using the reporter email

- **Exploited vs. Not Exploited Reports -** we identify the exploited reports using the snowball approach

- **Original vs. Duplicate Reports -** we identify the original reports using the bug status of the report

- **First Reported Time, Fixed Time, Released Time**

The entire cleaning process is described in detail in the paper

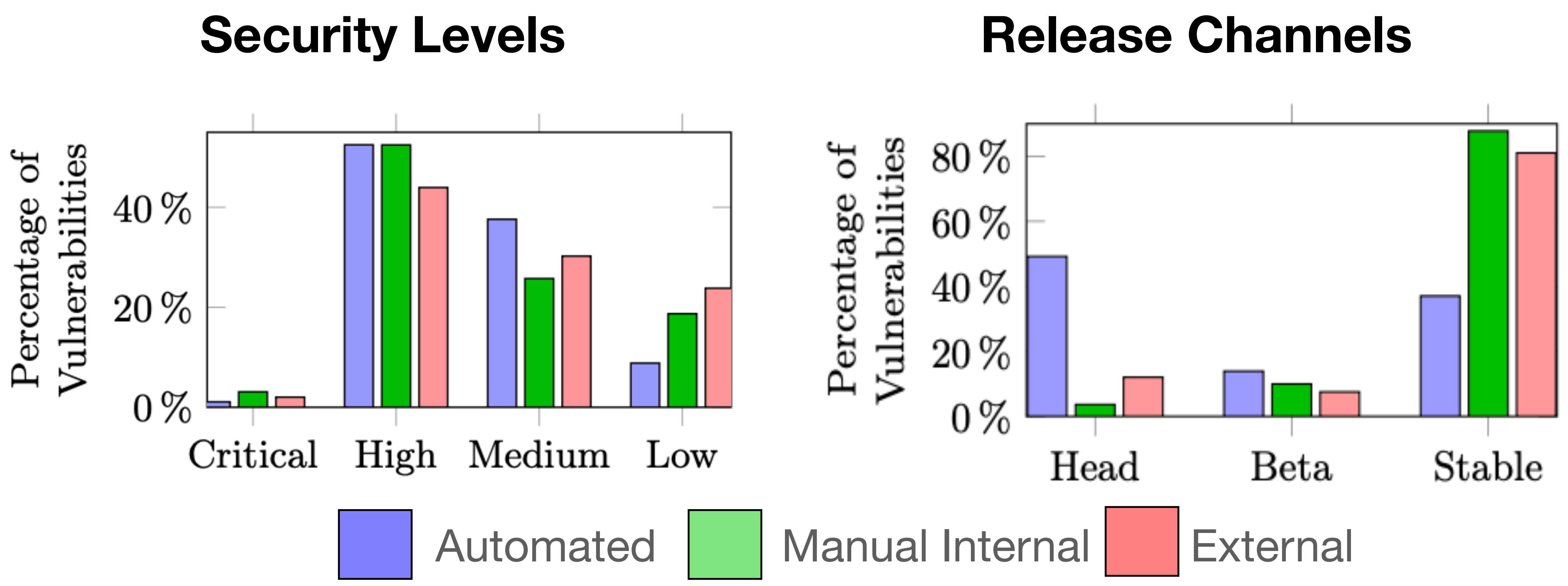
Research Question 1

Internal vs. External Discovery

*We observe **significant** differences between the type of vulnerabilities reported by external bug hunters and internal security teams*

External bug hunters focused more on reporting vulnerabilities

- impacting **head** release channels
- with **medium** and **low** severity
- containing specific weakness types (e.g., **Memory Buffer Bounds Error**)
- affecting the **User Interface** component
- where code base uses **C++** language

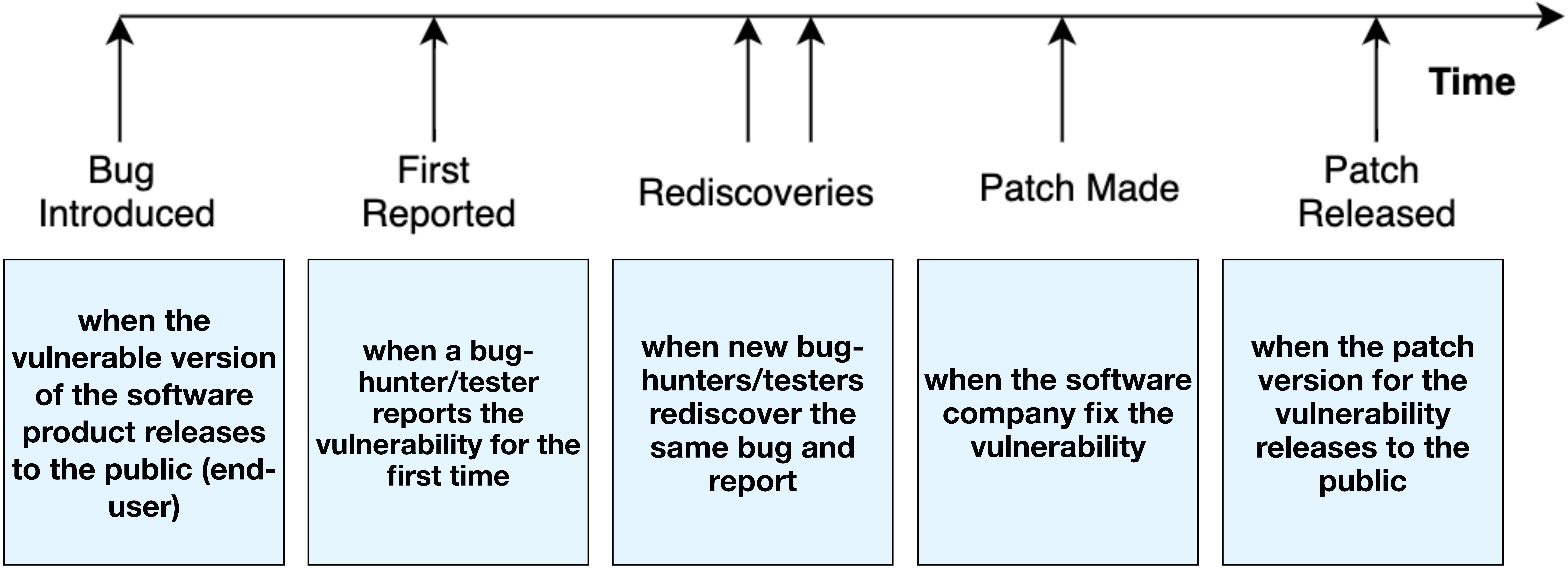


Chi-Squared Test Results on Distributions of Externally vs. Manually Internally Reported Valid Security Issues

Category	DOF	χ^2
Release Channel	2	143.11
Severity	3	75.52
Weakness Types	36	129.35
Components	471	2627.23
Programming Languages	17	132.96

Research Question 2

Rediscovery



Research Question 2

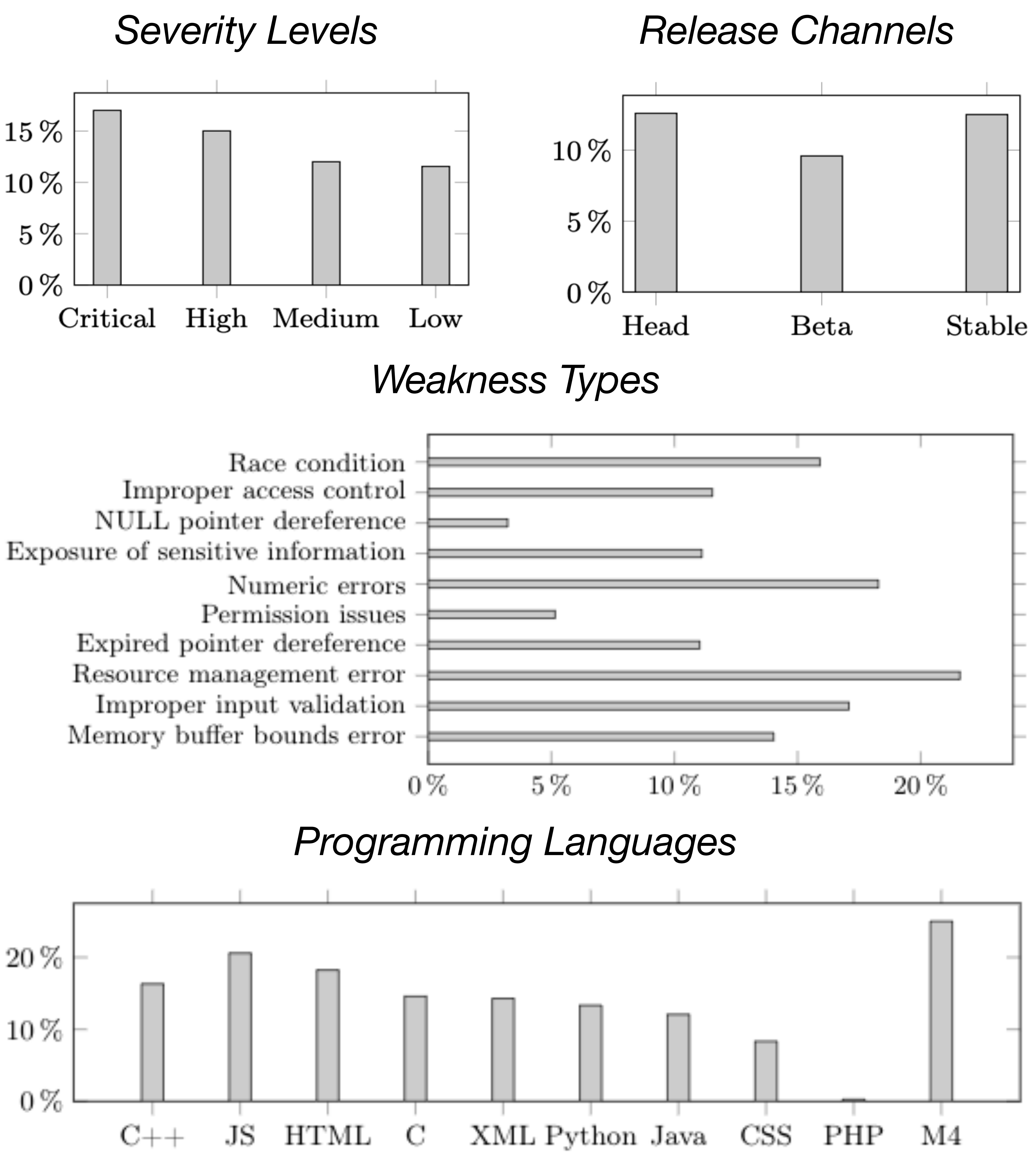
Rediscovery

*We observe that **rediscovery is non-negligible** and **certain** type of vulnerabilities are more likely to rediscover than other vulnerabilities*

Rediscoveries are more likely to occur in vulnerabilities

- containing specific weakness types (e.g., CWE 399: **Improper Management of System Resources**)
- affecting the **Rendering Engine (Blink)** component
- where code base uses certain languages (e.g., **C++**)

Percentage of vulnerabilities that are rediscovered at least once

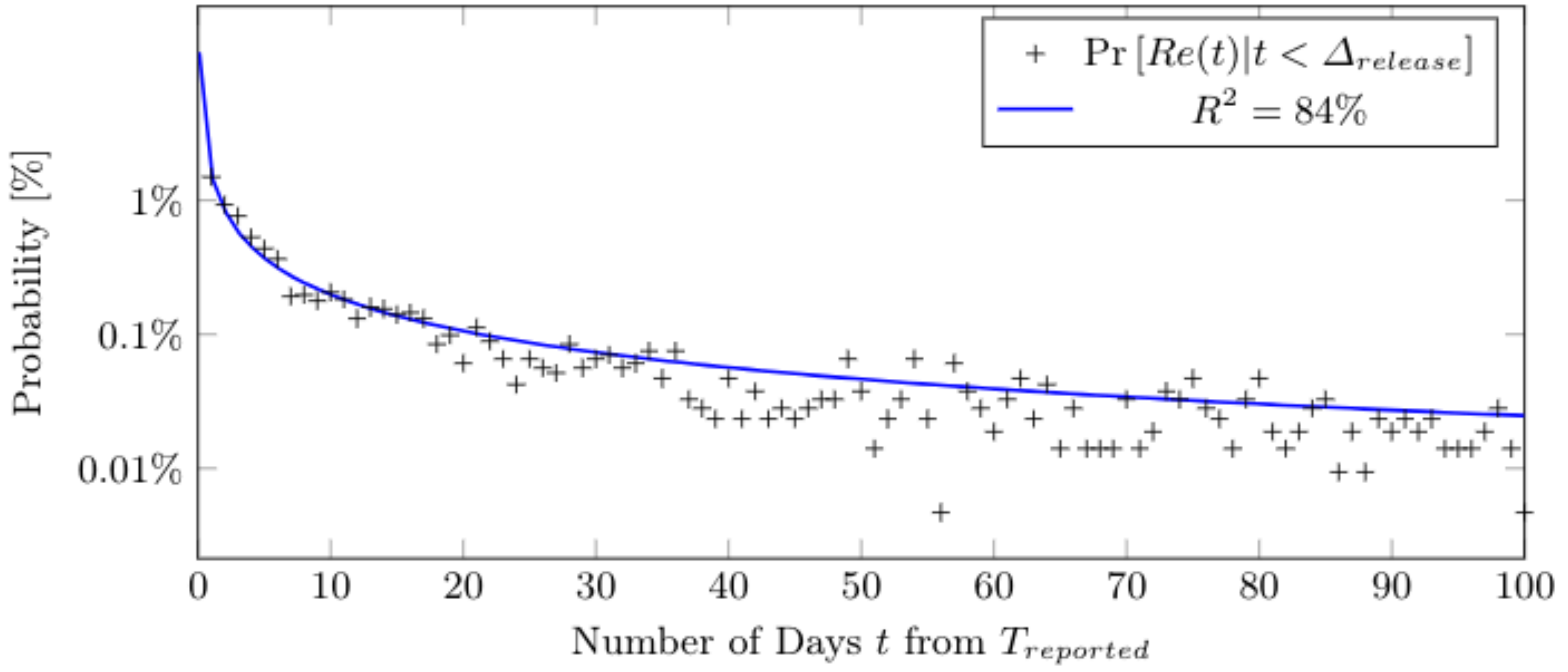


Research Question 2

Rediscovery

We observe that ***the probability of rediscovery decreases over time*** *from the first time a vulnerability is reported.*

Percentage of Vulnerabilities that are rediscovered on the t^{th} day after it is first reported

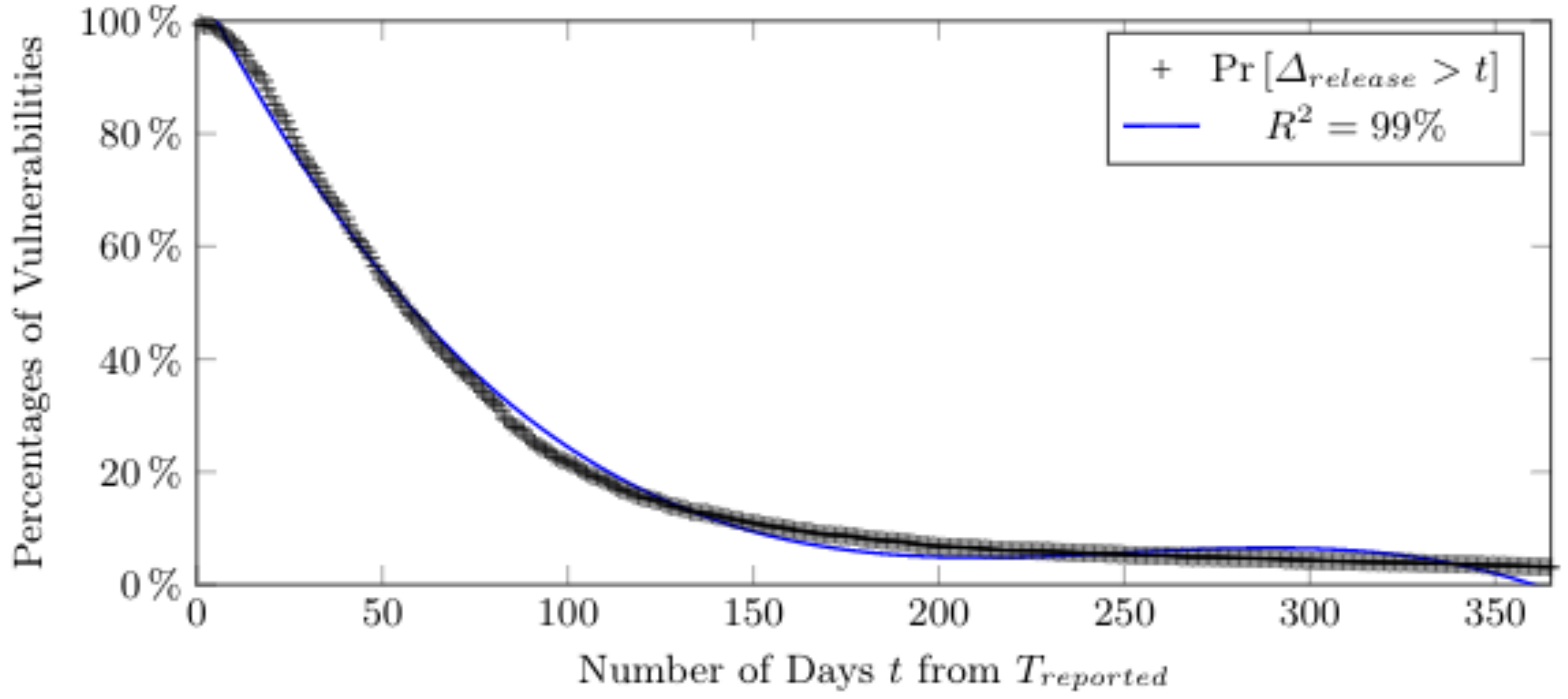


Fitted with Power Function: $2.032\ t^{-1.058}$

Further, half of the all reported vulnerabilities

- are **fixed within 8 days** from when they are first reported
- are **patched within 55 days** from when they are first reported

Percentage of Vulnerabilities that are not patched in t days after it is first reported



Fitted with Polynomial Function: $-7 .10^{-8}\ t^3 + 5\ .\ 10^{-5}\ t^2 + 0.0128\ t + 1.0668$

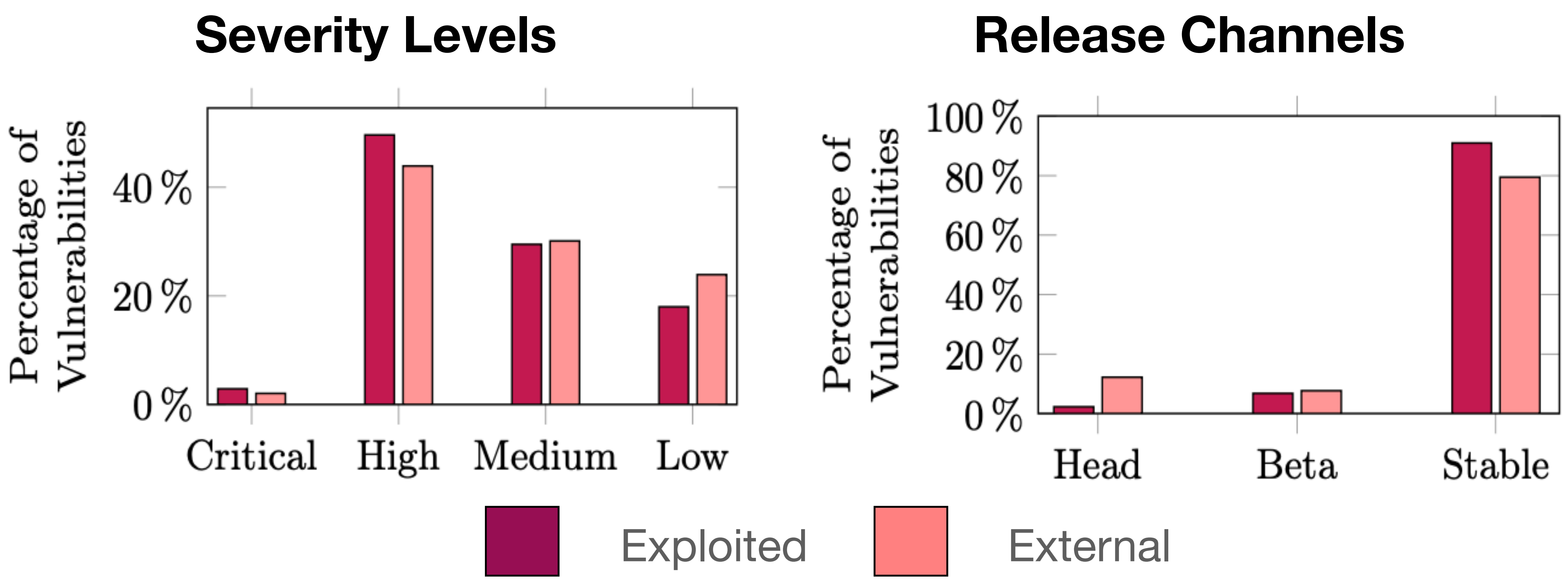
Research Question 3

External Discovery vs. Exploited in the Wild

*We observe **significant** differences between the type of vulnerabilities exploited in the wild and reported by external bug hunters*

Threat actors focused more on exploiting vulnerabilities

- impacting **stable** release channels
- with **critical** and **high** severity
- affecting the **Rendering Engine (Blink)** component
- containing specific weakness types (e.g., **CWE 399: Improper Management of System Resources**)
- where code base uses **C++** language



Chi-Squared Test Results on the Distributions of Exploited Issues vs. All other Externally Reported Issues

Category	DOF	χ^2
Release Channel	2	53.01
Severity	3	1.6
Components	113	691.83
Weakness Types	6	30.08
Programming Languages	18	45.29

Conclusion and Future Work

- External bug hunters with their diverse expertise provide security benefits by **complementing the internal security teams**.
- The Probability of vulnerability rediscovery is **non-negligible,** and it varies based on different inherent features of the vulnerability report.
  - **Should the chromium team put more focus on the types of vulnerabilities that are more likely to rediscovered ?**
- The exploited in the wild analysis shows that the type of vulnerabilities exploited by threat actors significantly differs from those reported by external bug hunters.
  - **Should the chromium team shift the focus of vulnerability-discovery efforts towards types of vulnerabilities that are exploited more often ?**
- In future work,
  - we plan to extend our study with **another software product**, such as Firefox.
  - we aim to develop a model for **quantifying the benefits of vulnerability discovery and patching.**

Thank You For The Attention !

Q & A

Amutheezan Sivagnanam
Email: asivagnanam@uh.edu
Homepage: https://amutheezan.com/

Aron Laszka
Email: alaszka@uh.edu
Homepage: https://aronlaszka.com/

University of Houston
Amutheezan Sivagnanam, Soodeh Atefi, Afiya Ayman and Aron Laszka

Technical University of Munich
Jens Grossklags

[this slide is kept blank intentionally]

Additional Slides

Research Question 1

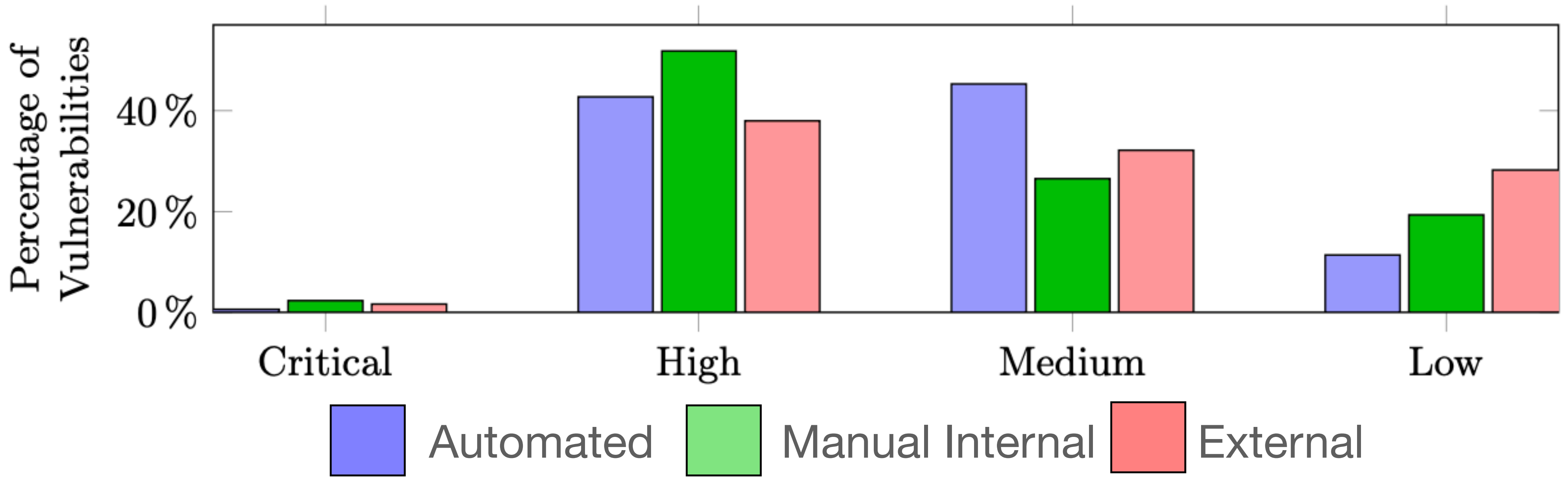
Internal vs. External Discovery (Stable Release Channel)

*We observe **significant** differences between the type of vulnerabilities reported by external bug hunters and internal security teams*

External bug hunters focused more on reporting vulnerabilities

- with **medium** and **low** severity
- containing specific weakness types (e.g., **Memory Buffer Bounds Error**)
- affecting the **User Interface** component
- where code base uses **C++** language

Severity Levels



Chi-Squared Test Results on Distributions of Externally vs. Manually Internally Reported Valid Security Issues

Category	DOF	χ^2
Severity	3	115.27
Weakness Types	36	123.52
Components	382	1364.62
Programming Languages	17	97.06

Research Question 3

External Discovery vs. Exploited in the Wild

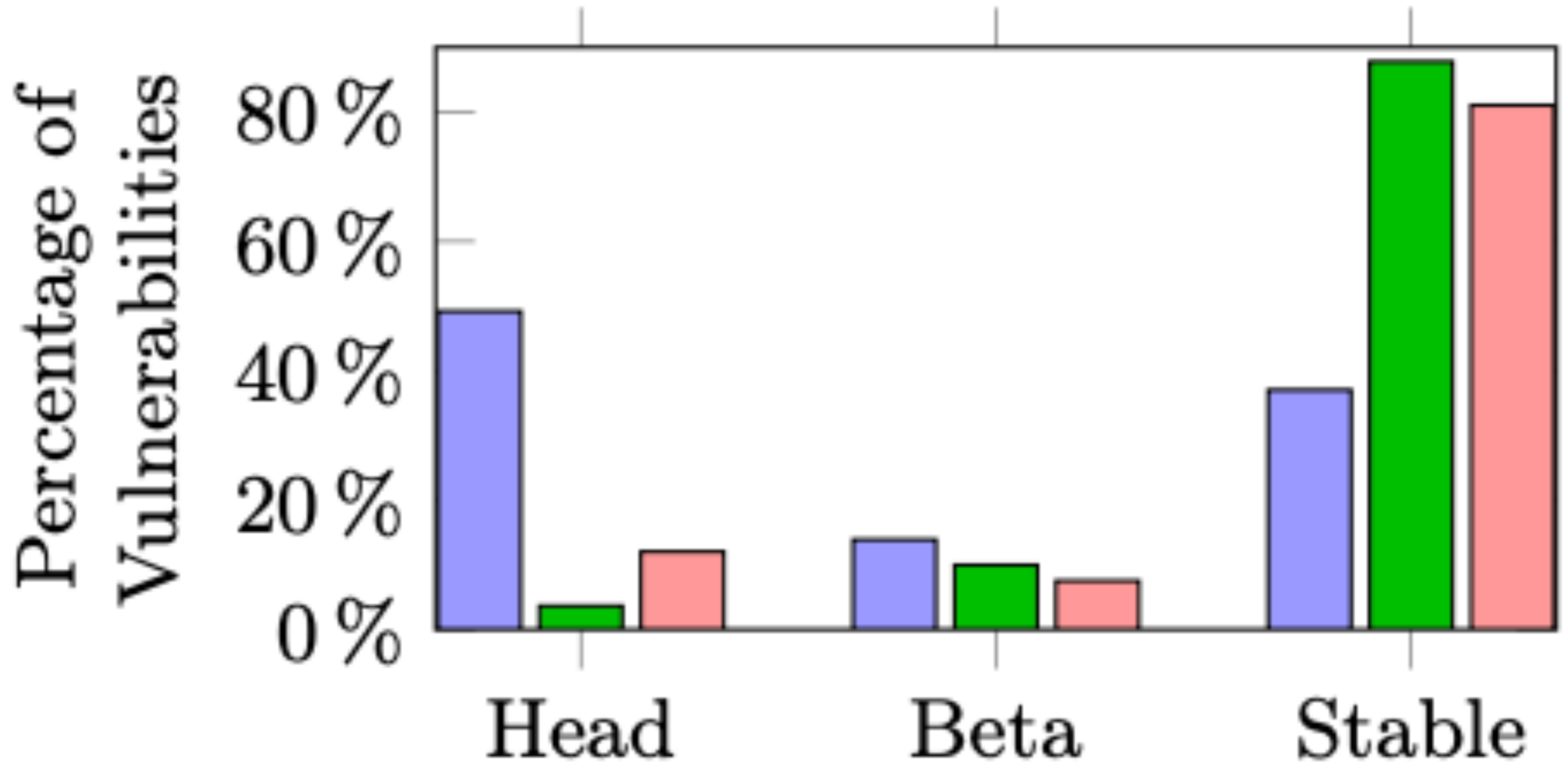
- We Identify exploited issues using **snowball** approach
- In the snowball approach
  - **Step 1**: Collect the issues that contain the phrase "**exploited in the wild**" in the description
  - **Step 2**: Identify the other relative terms from collected issues, repeat the Step 1, with the new terms we identified in this step.
  - **Step 3:** If no new words obtain at the end of Step 2, then stop the process

Exploited in the wild
Exploit in the wild
zero day
zero-day
out in the wild
occurring in the wild

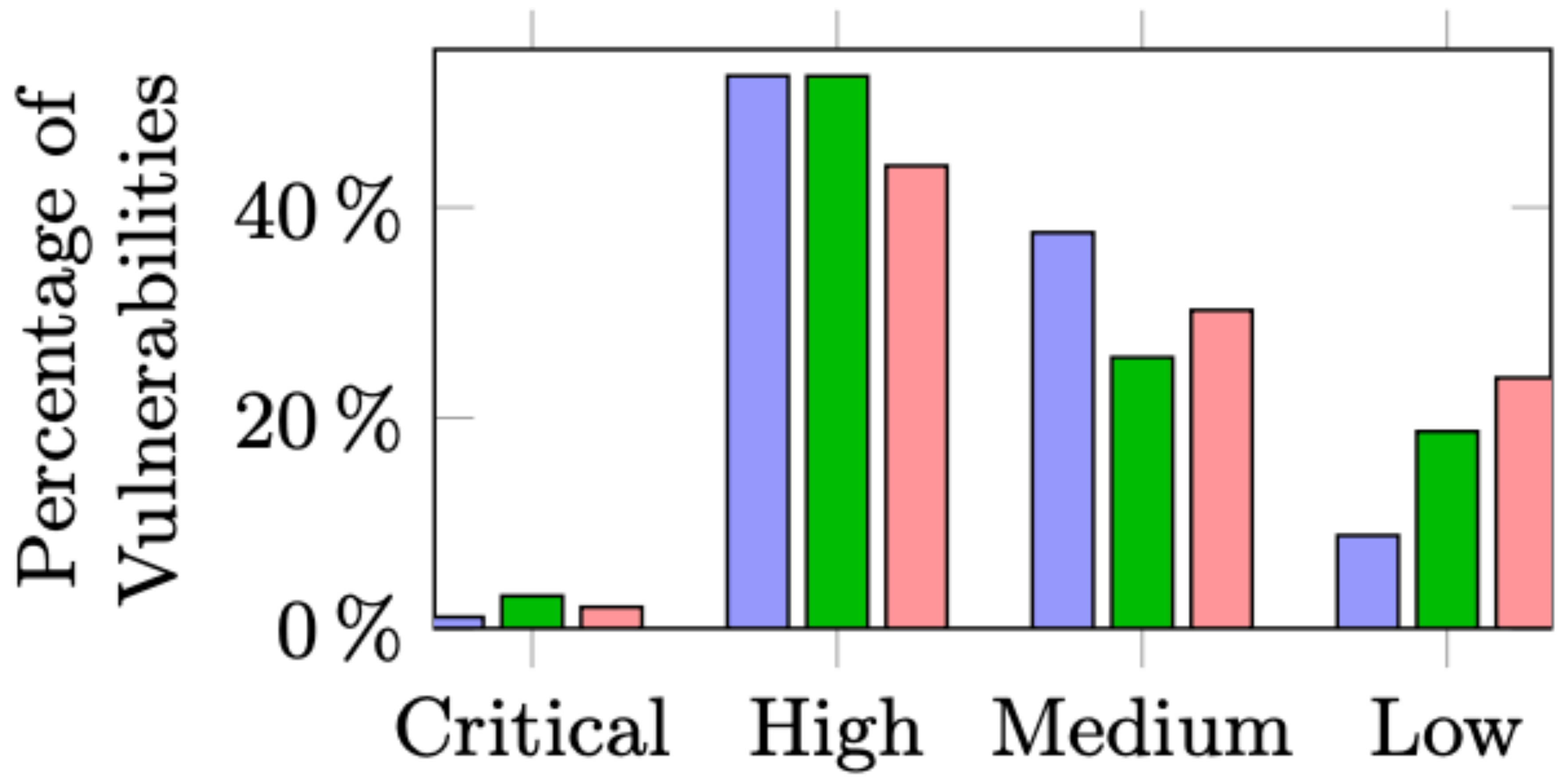
Sample terms used in the snowball approach

Research Question 1

Additional Results



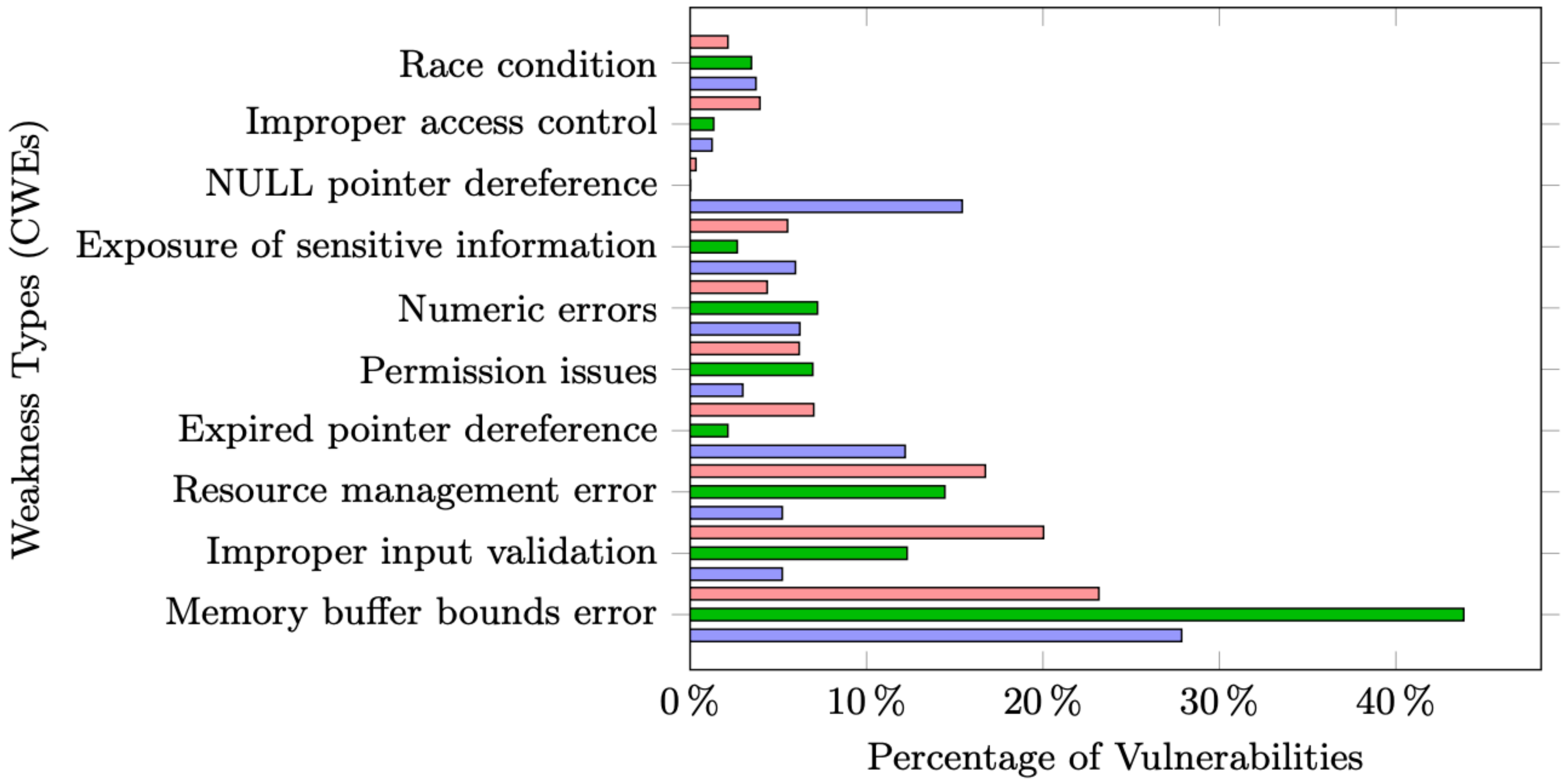
(a) Release Channel



(b) Security Severity

Research Question 1

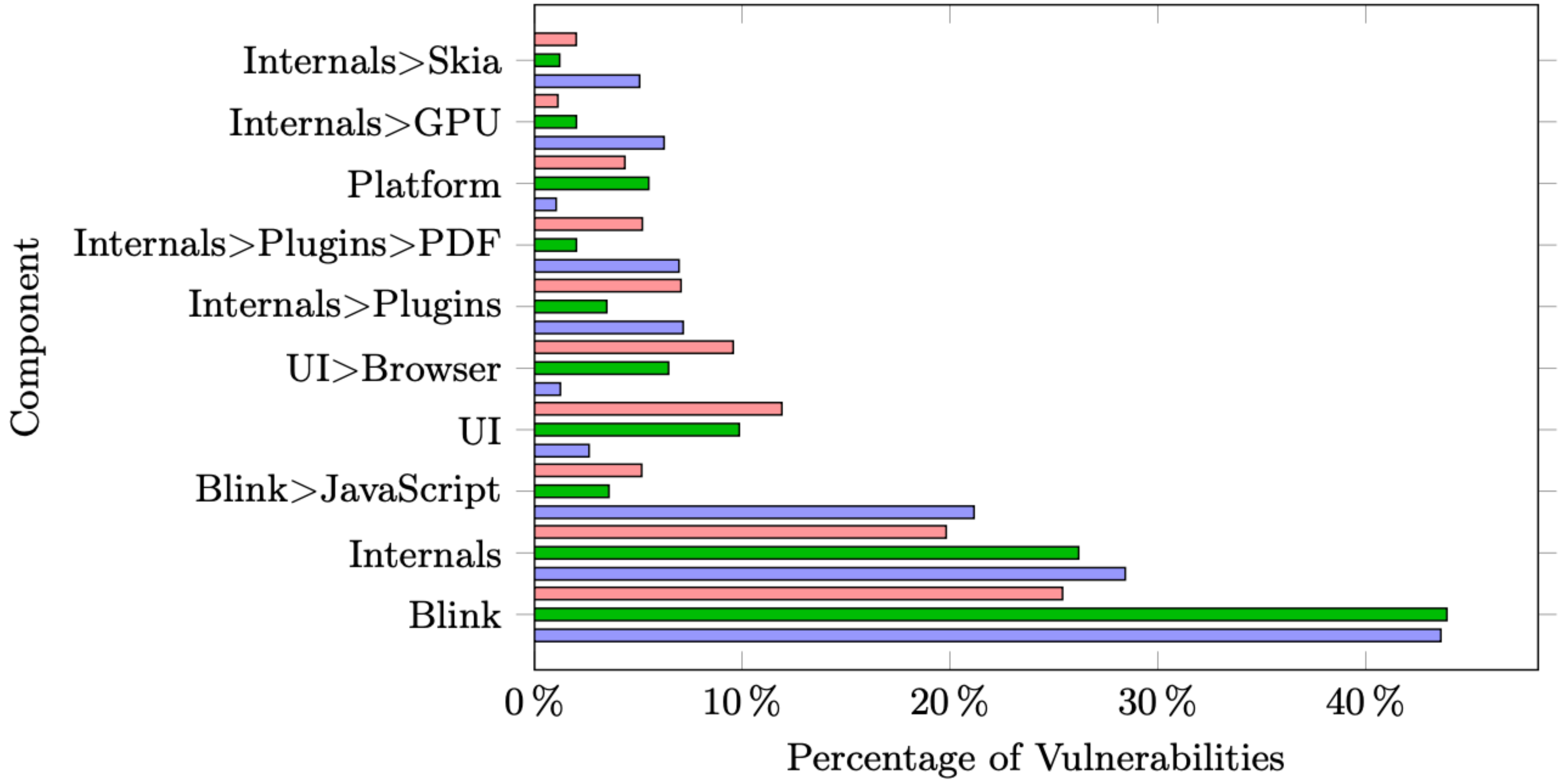
Additional Results



(c) Weakness Type

Research Question 1

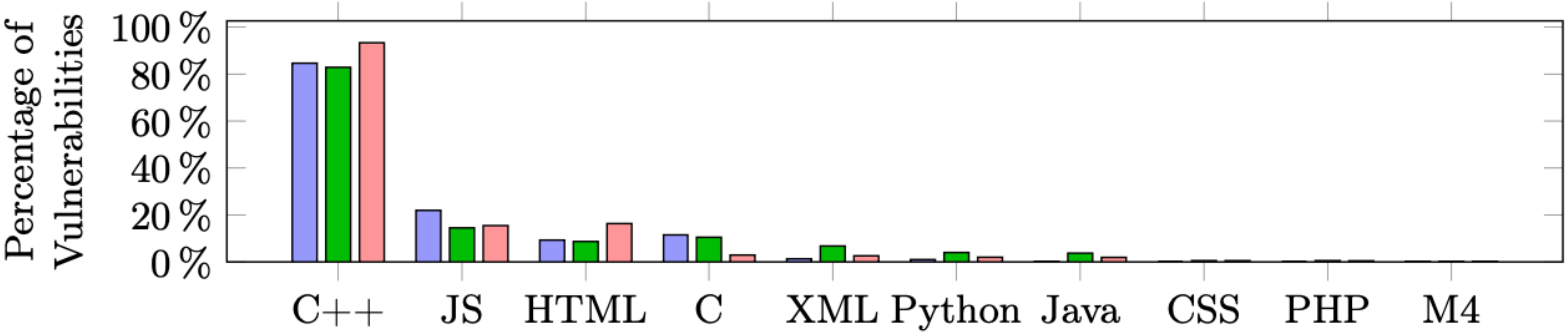
Additional Results



(d) Component

Research Question 1

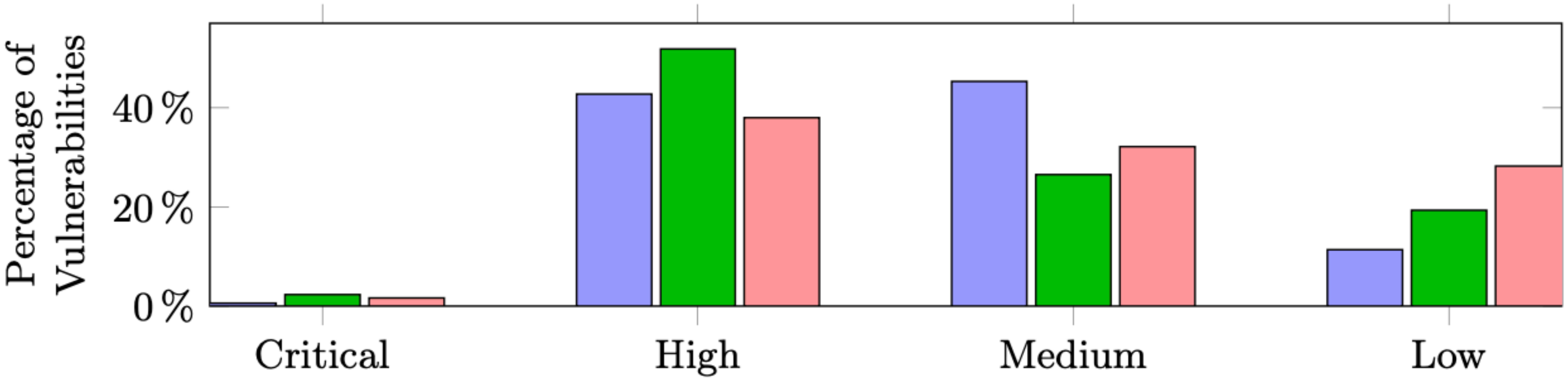
Additional Results



(e) Programming Language

Research Question 1

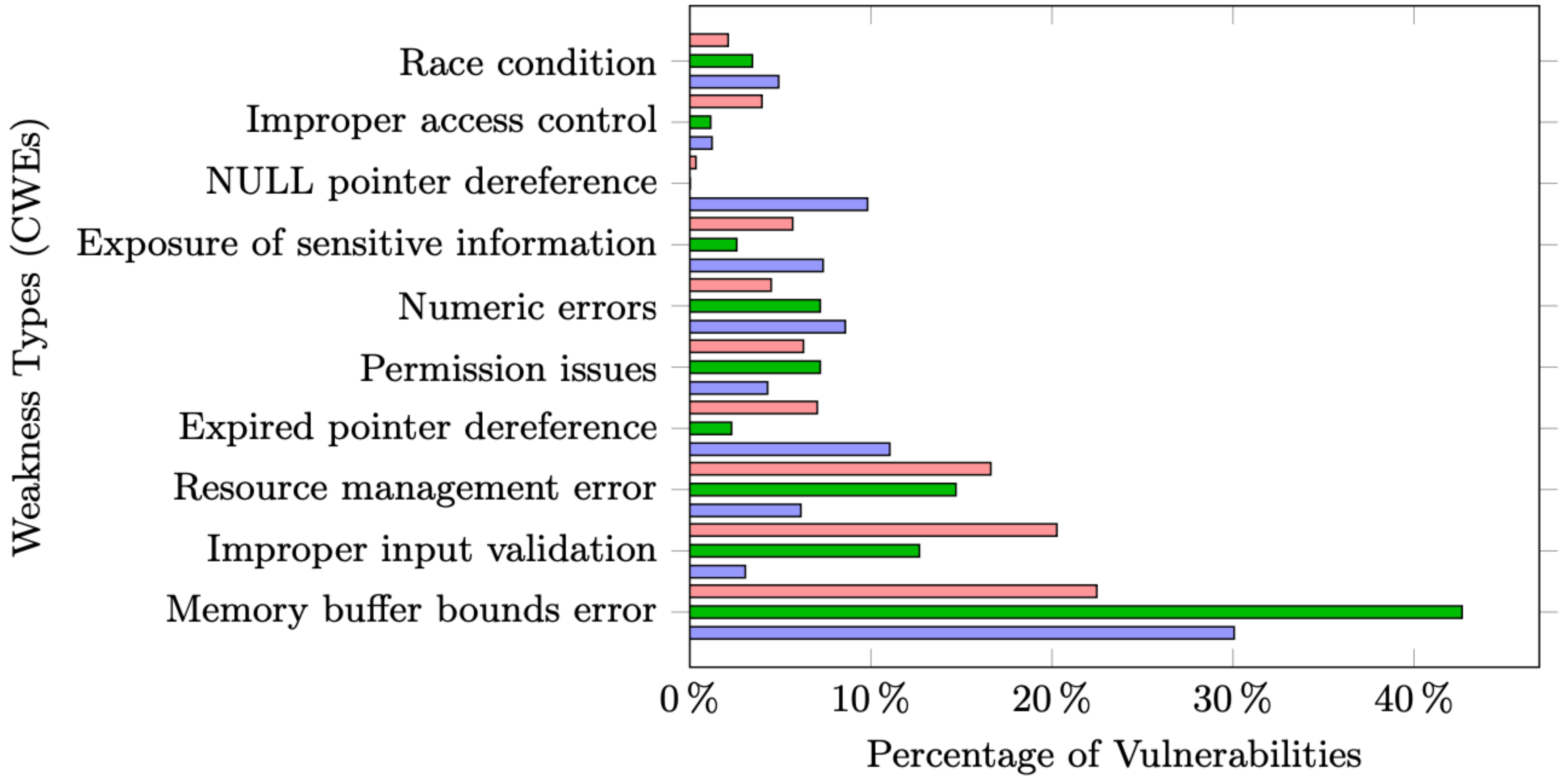
Additional Results



(a) Security Severity

Research Question 1

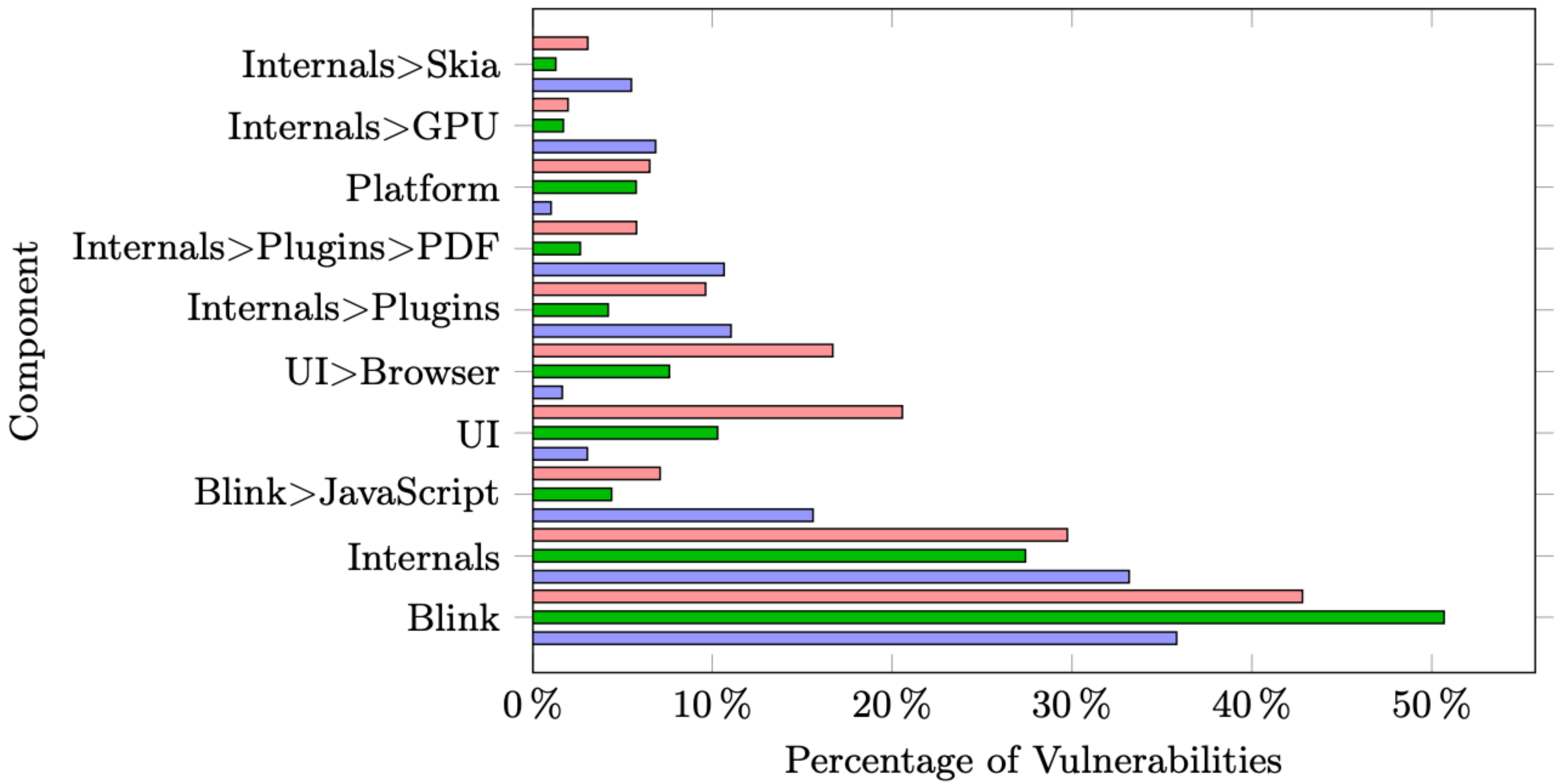
Additional Results



(b) Weakness Type

Research Question 1

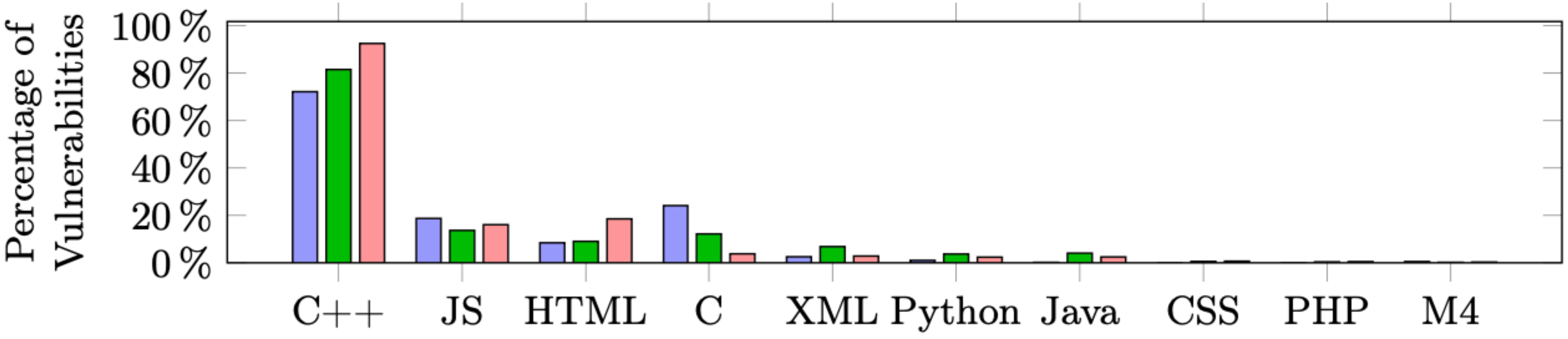
Additional Results



(c) Component

Research Question 1

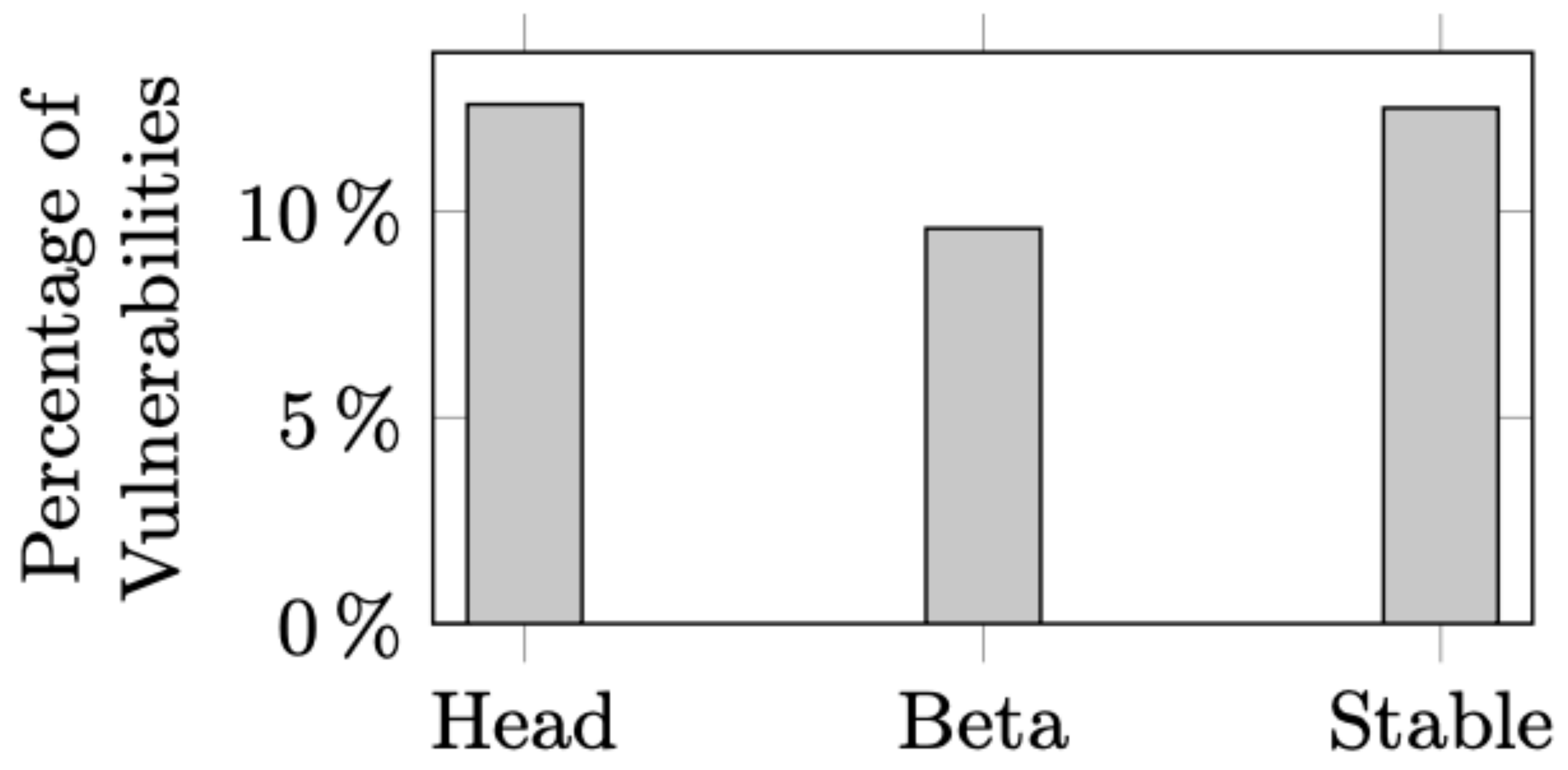
Additional Results



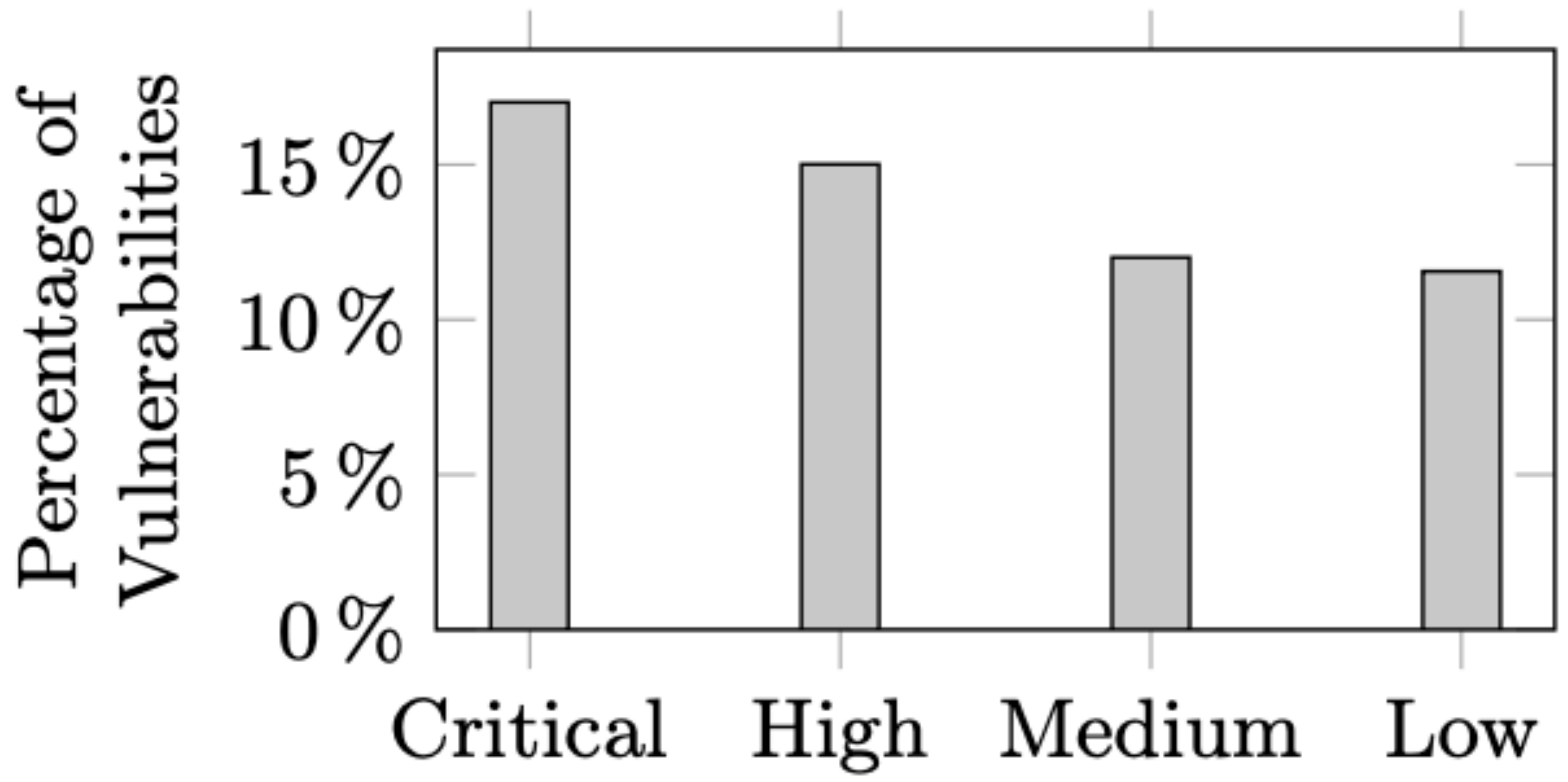
(d) Programming Language

Research Question 2

Additional Results



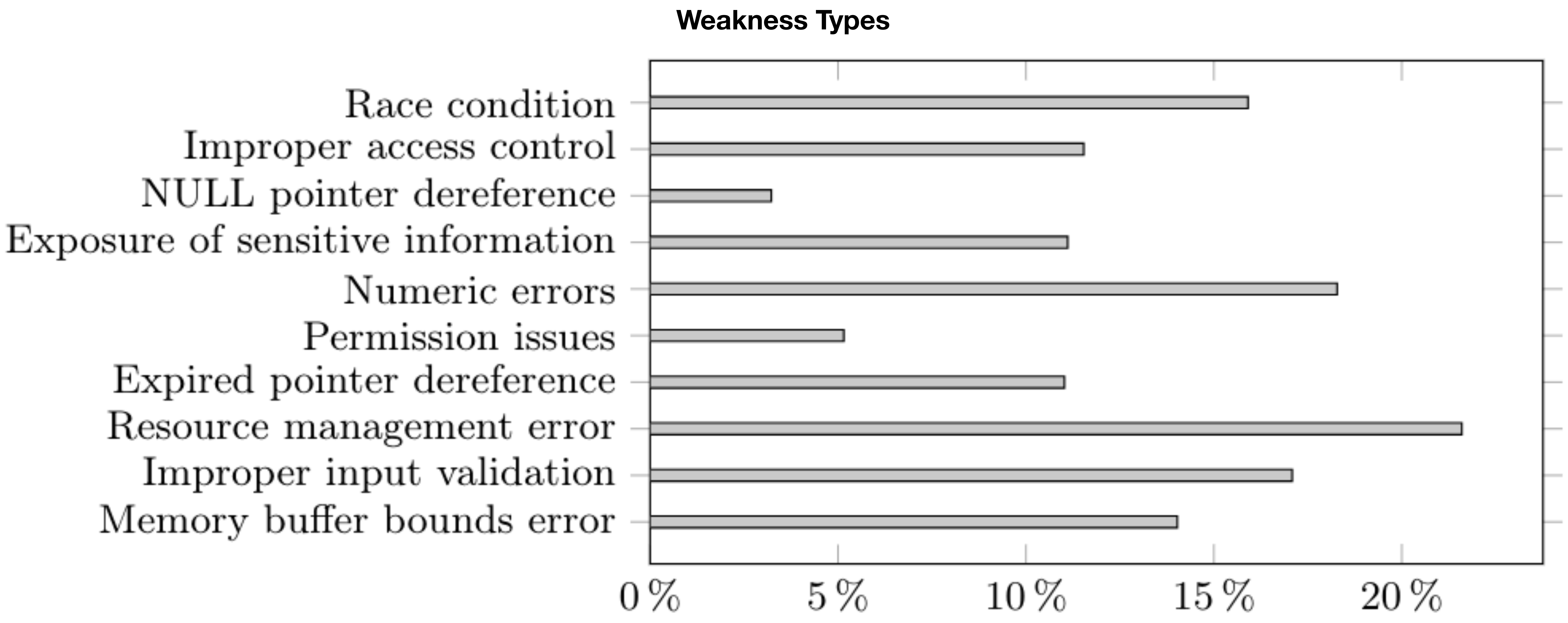
(a) Release Channel



(b) Security Severity

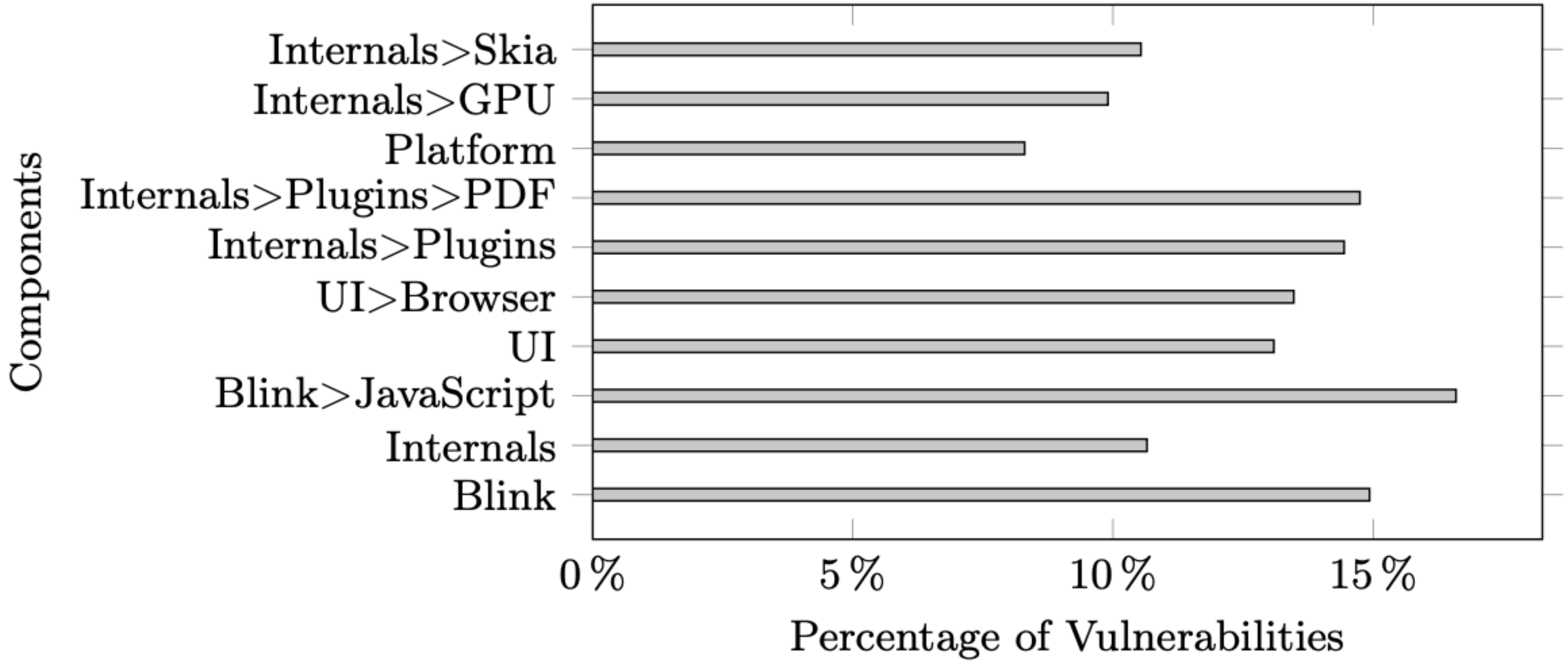
Research Question 2

Additional Results



Research Question 2

Additional Results

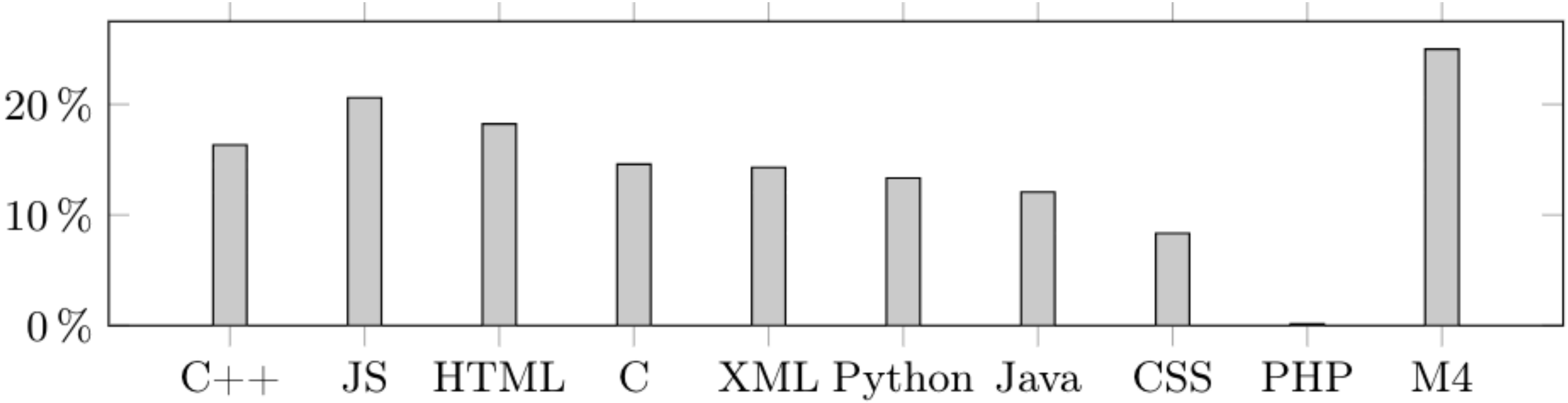


(d) Component

Research Question 2

Additional Results

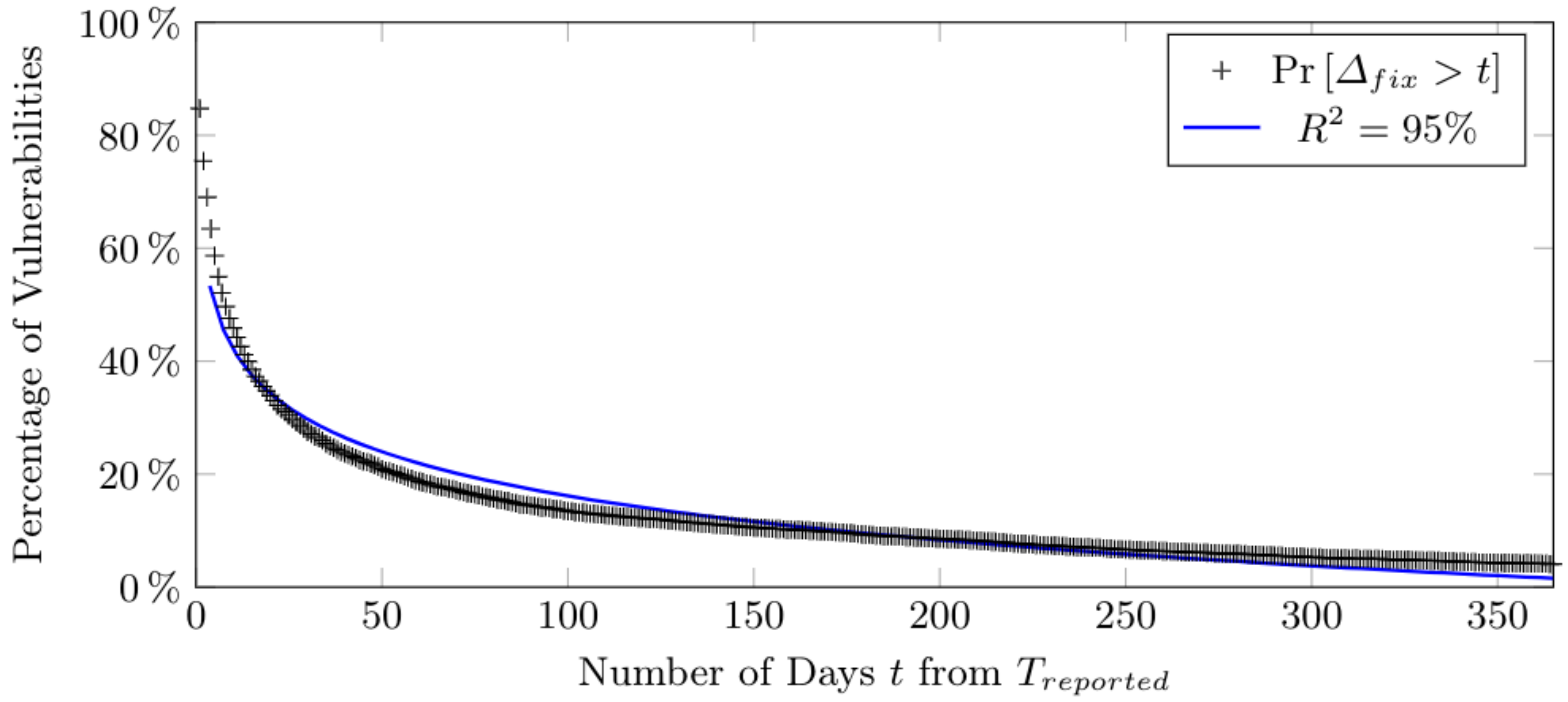
Programming Languages



Research Question 2

Additional Results

Percentage of Vulnerabilities that are not fixed in t days after it is first reported

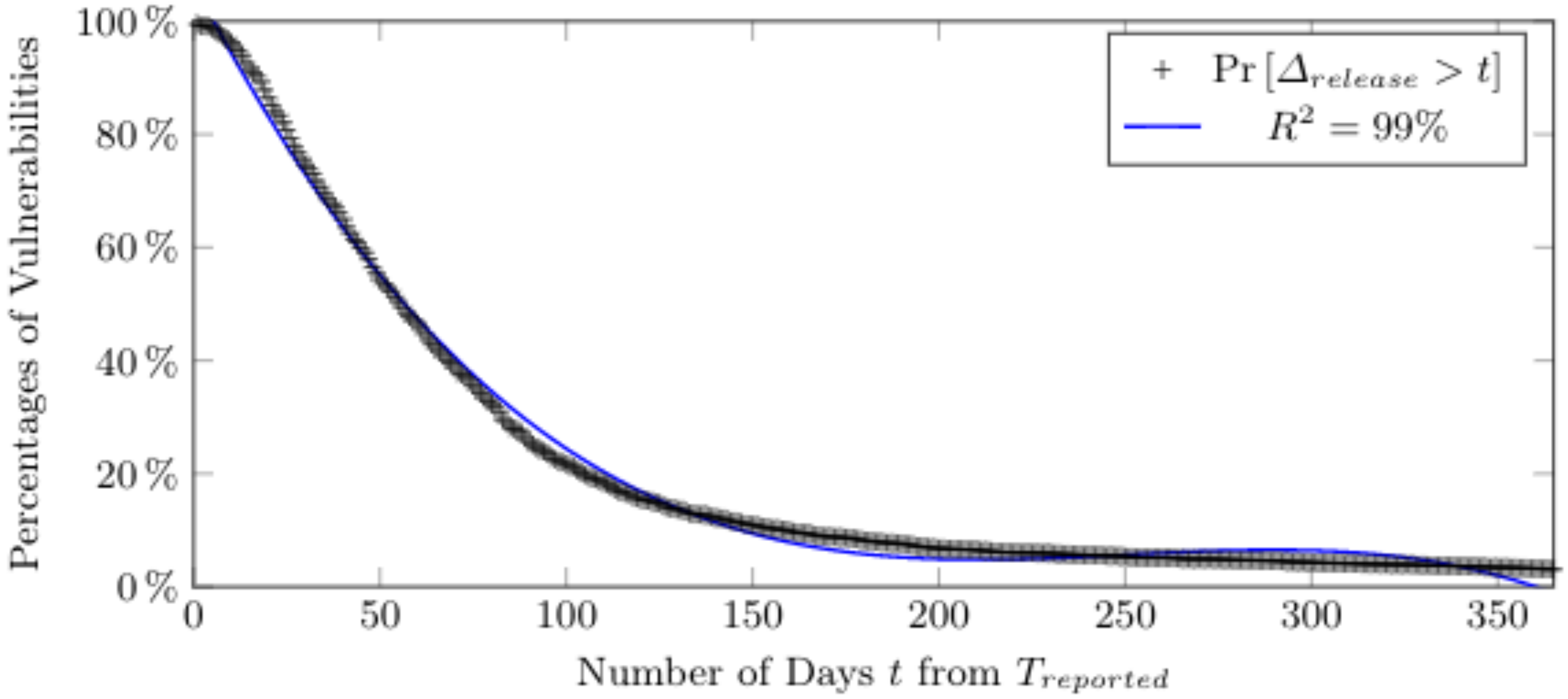


Fitted with Logarithamic Function: -0.113 ln (t) + 0.6805

Research Question 2

Additional Results

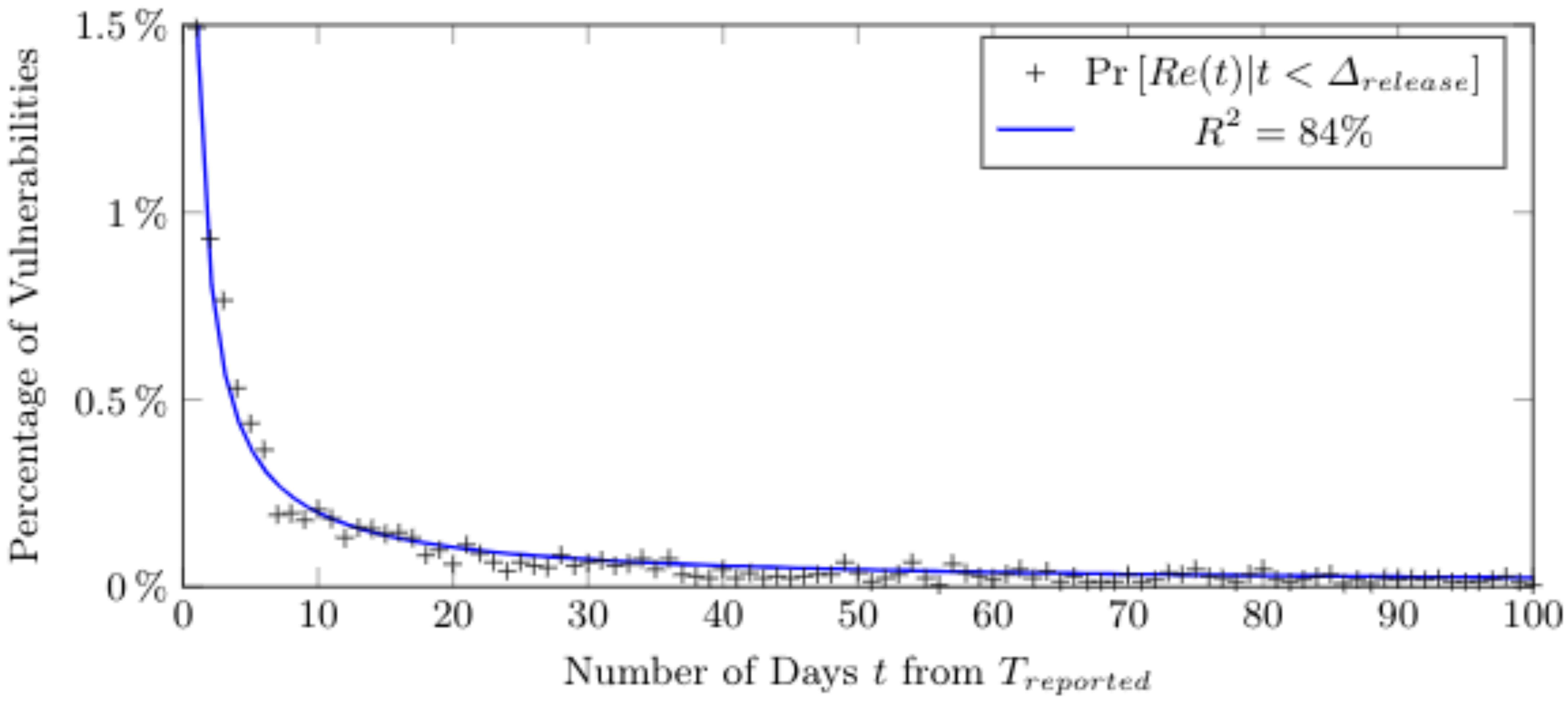
Percentage of Vulnerabilities that are not patched in t days after it is first reported



Fitted with Polynomial Function: -7 x 10^{-8} t^3 + 5 x 10^{-5} t^2 + 0.0128 t + 1.0668

Research Question 2
Additional Results

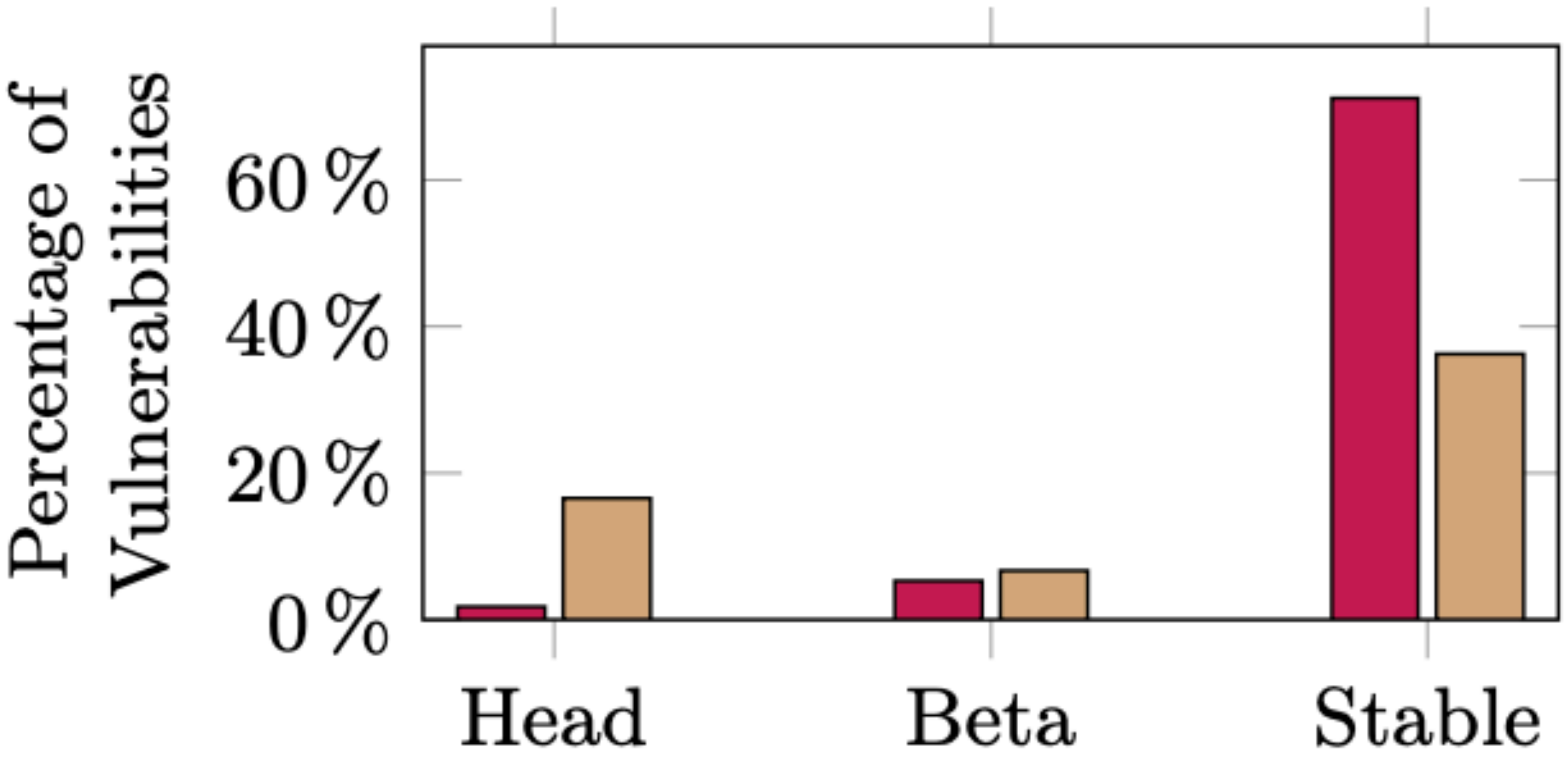
Percentage of Vulnerabilities that are rediscovered on the t^{th} day after it is first reported



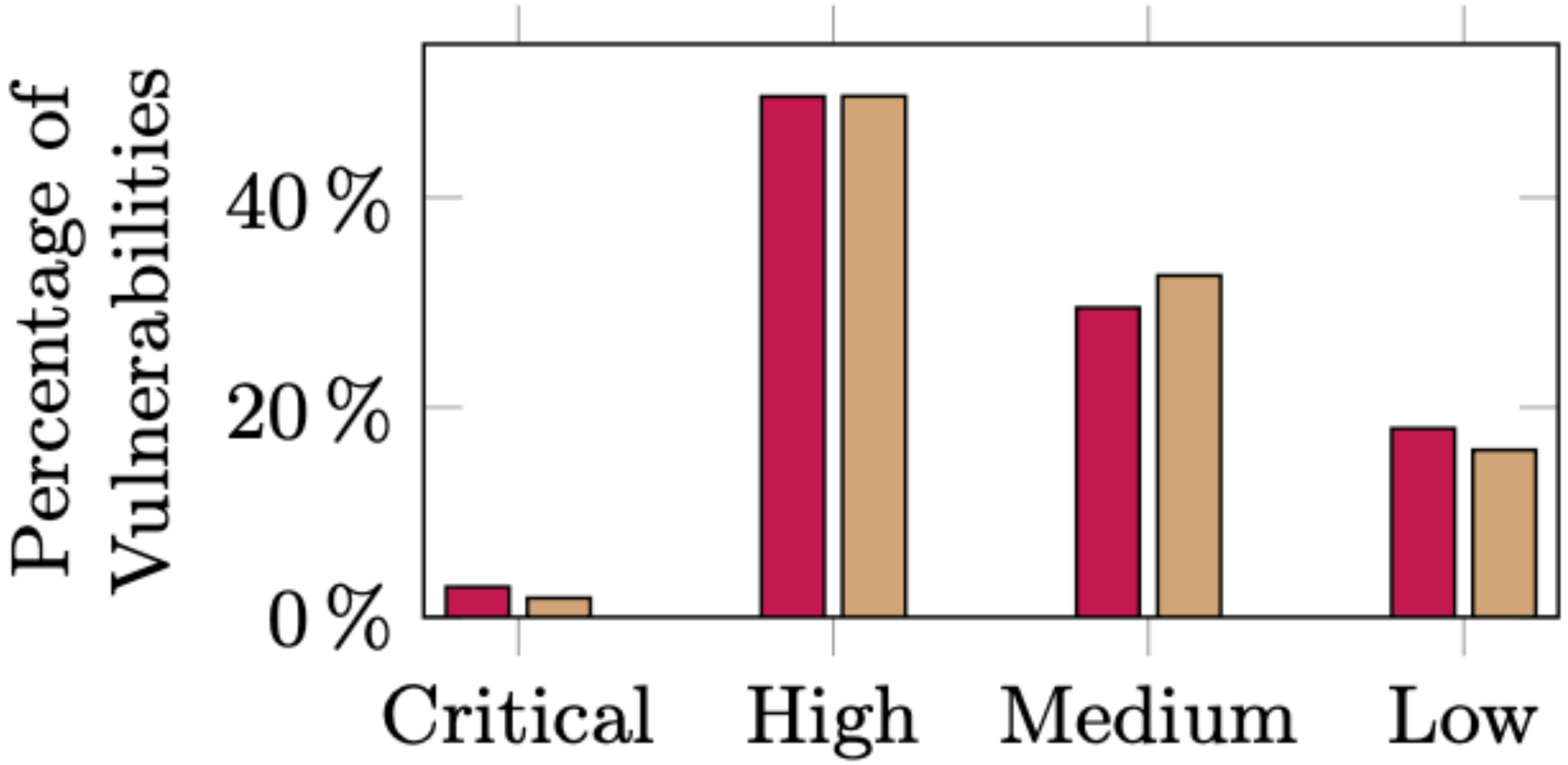
Fitted with Power Function: $2.032 \times t^{-1.058}$

Research Question 3

Additional Results



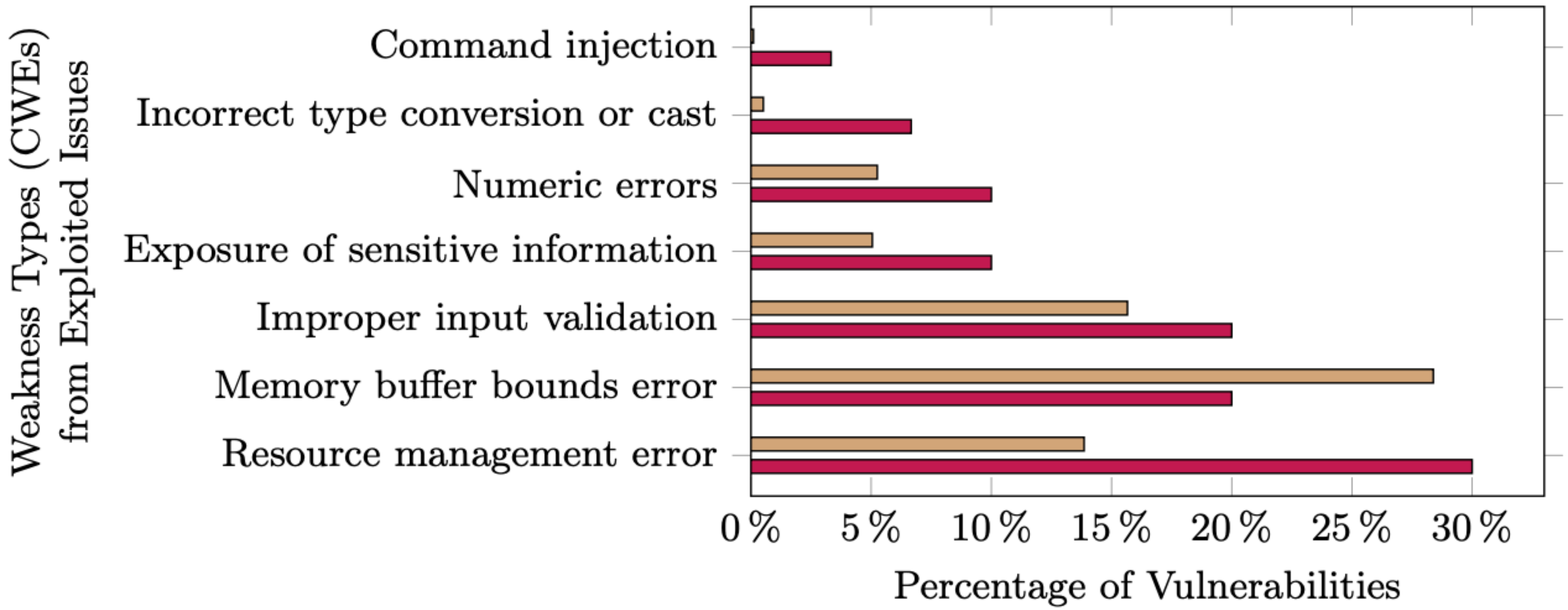
(a) Release Channel



(b) Security Severity

Research Question 3

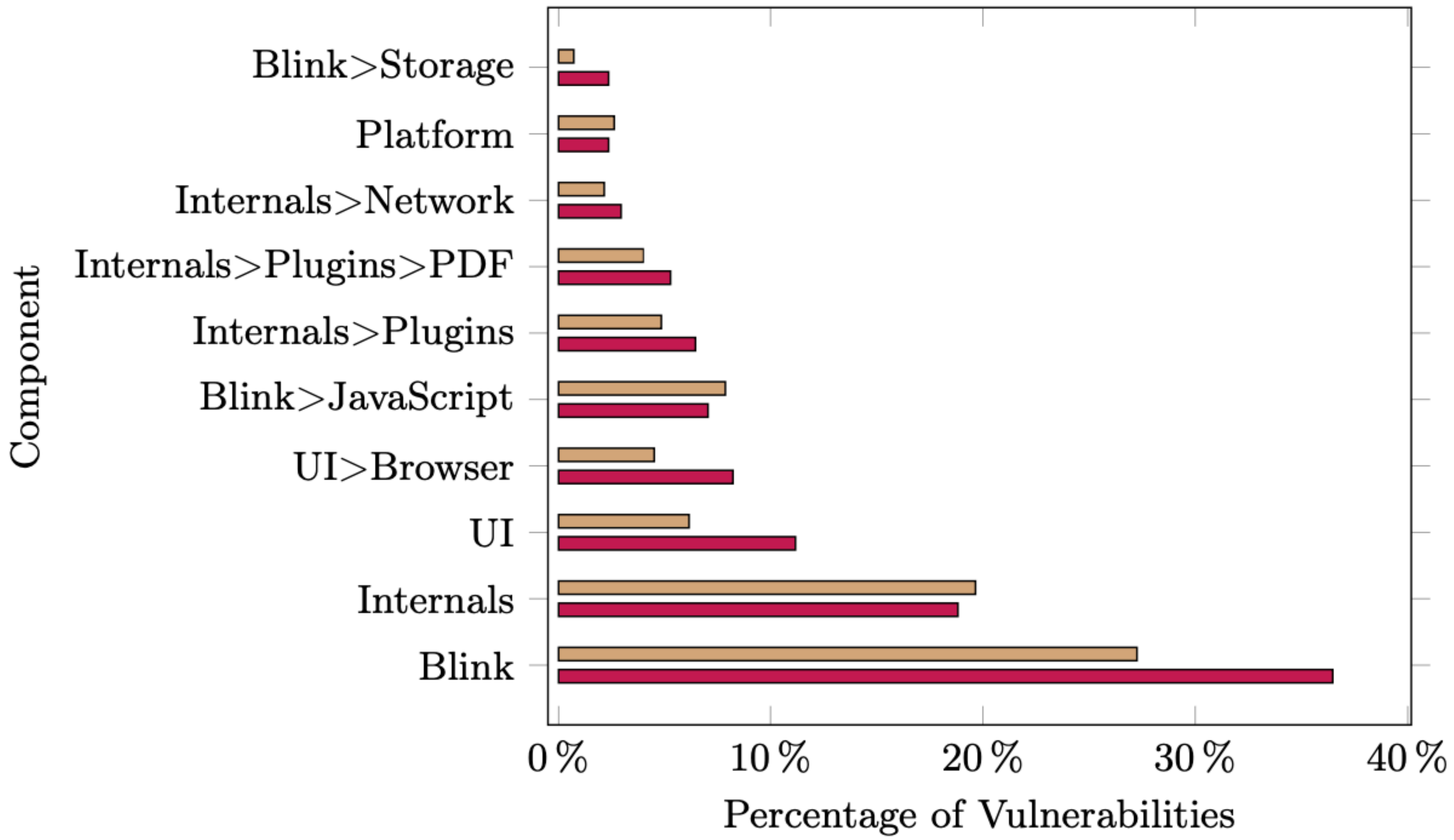
Additional Results



(c) Weakness Type

Research Question 3

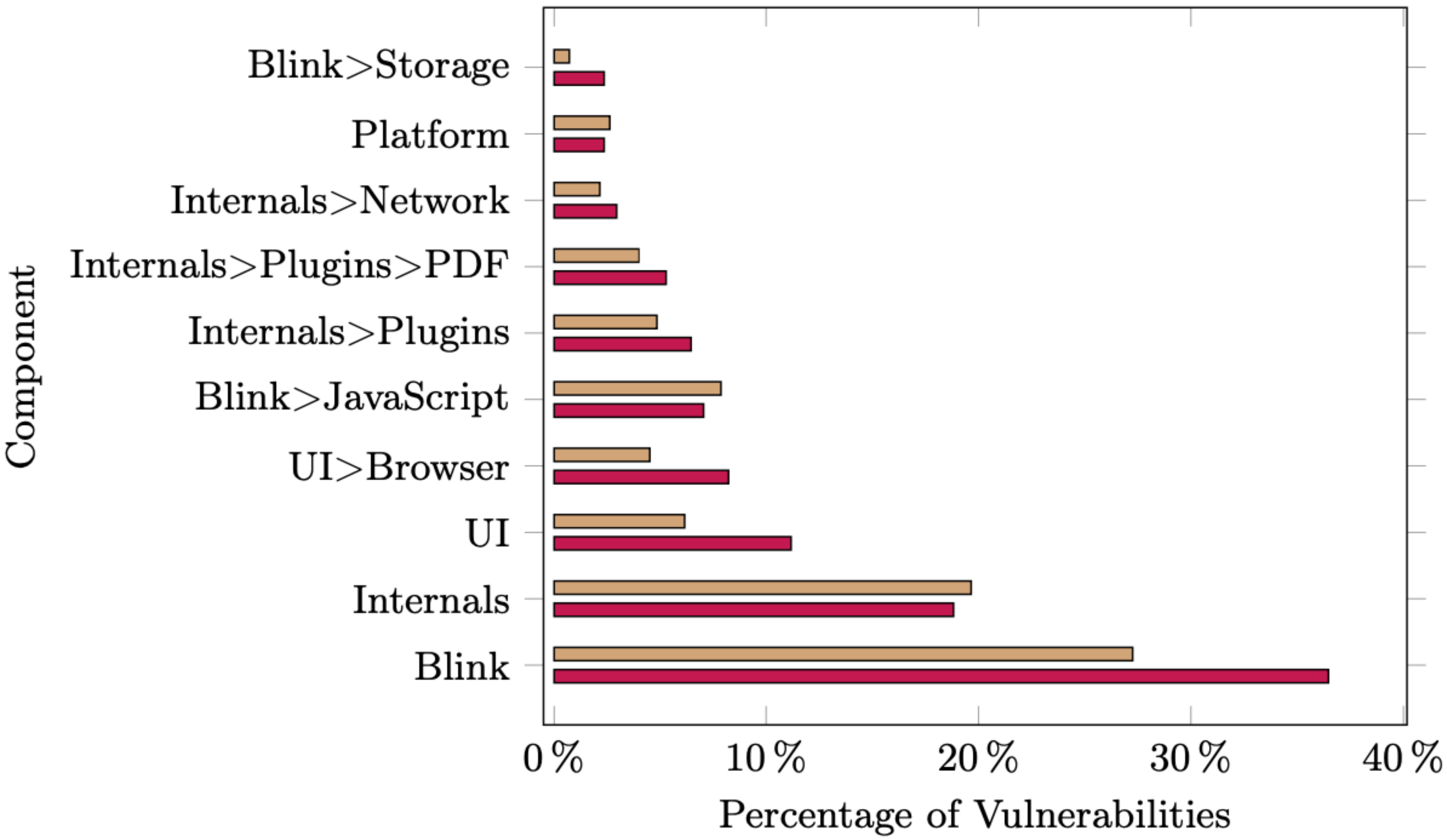
Additional Results



(d) Component

Research Question 3

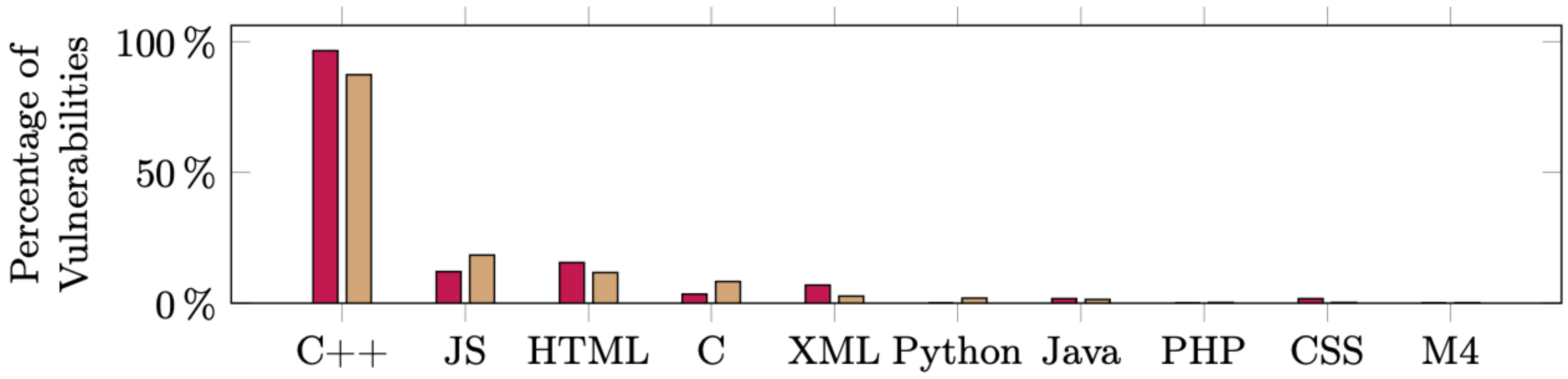
Additional Results



(d) Component

Research Question 3

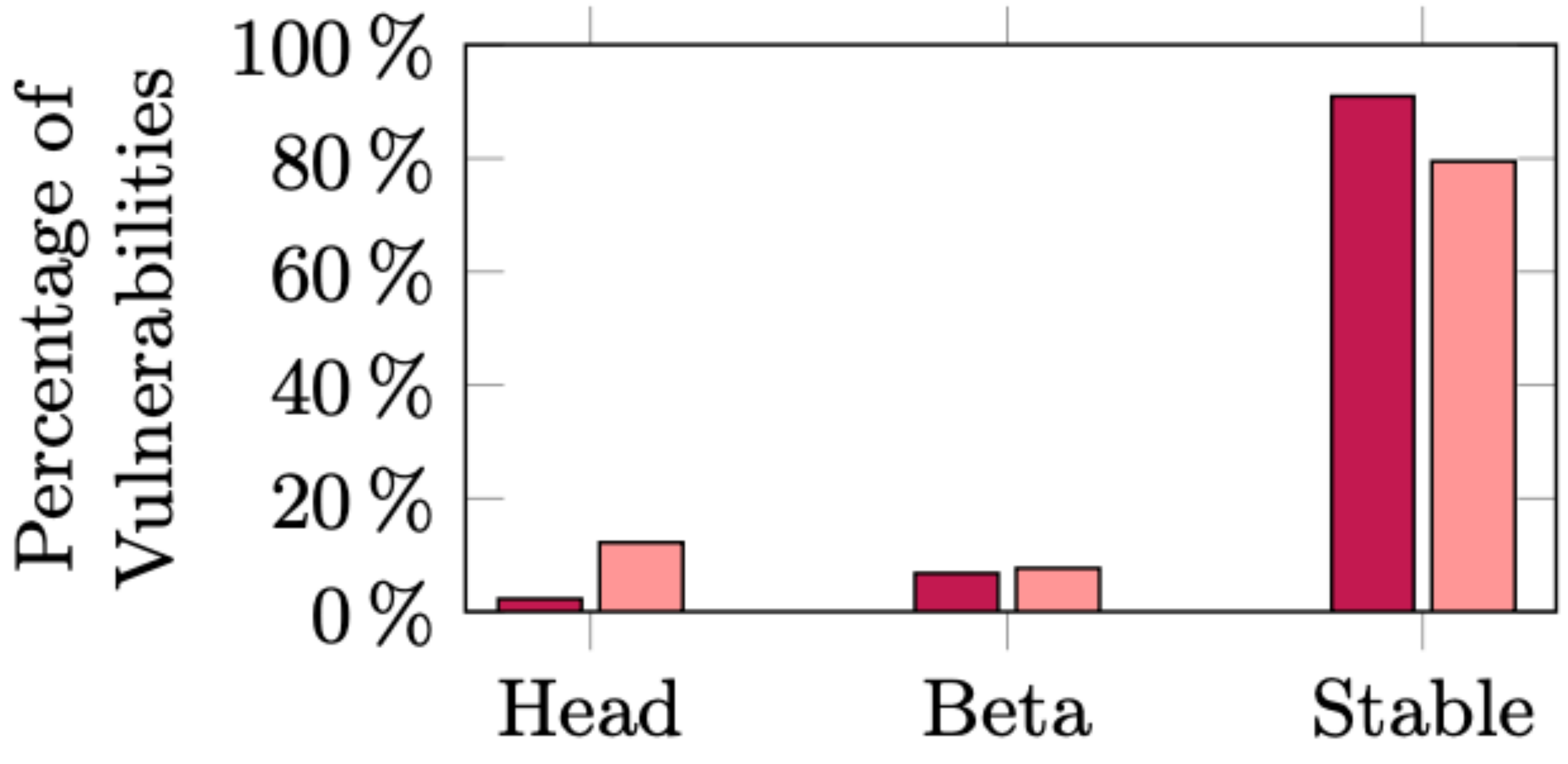
Additional Results



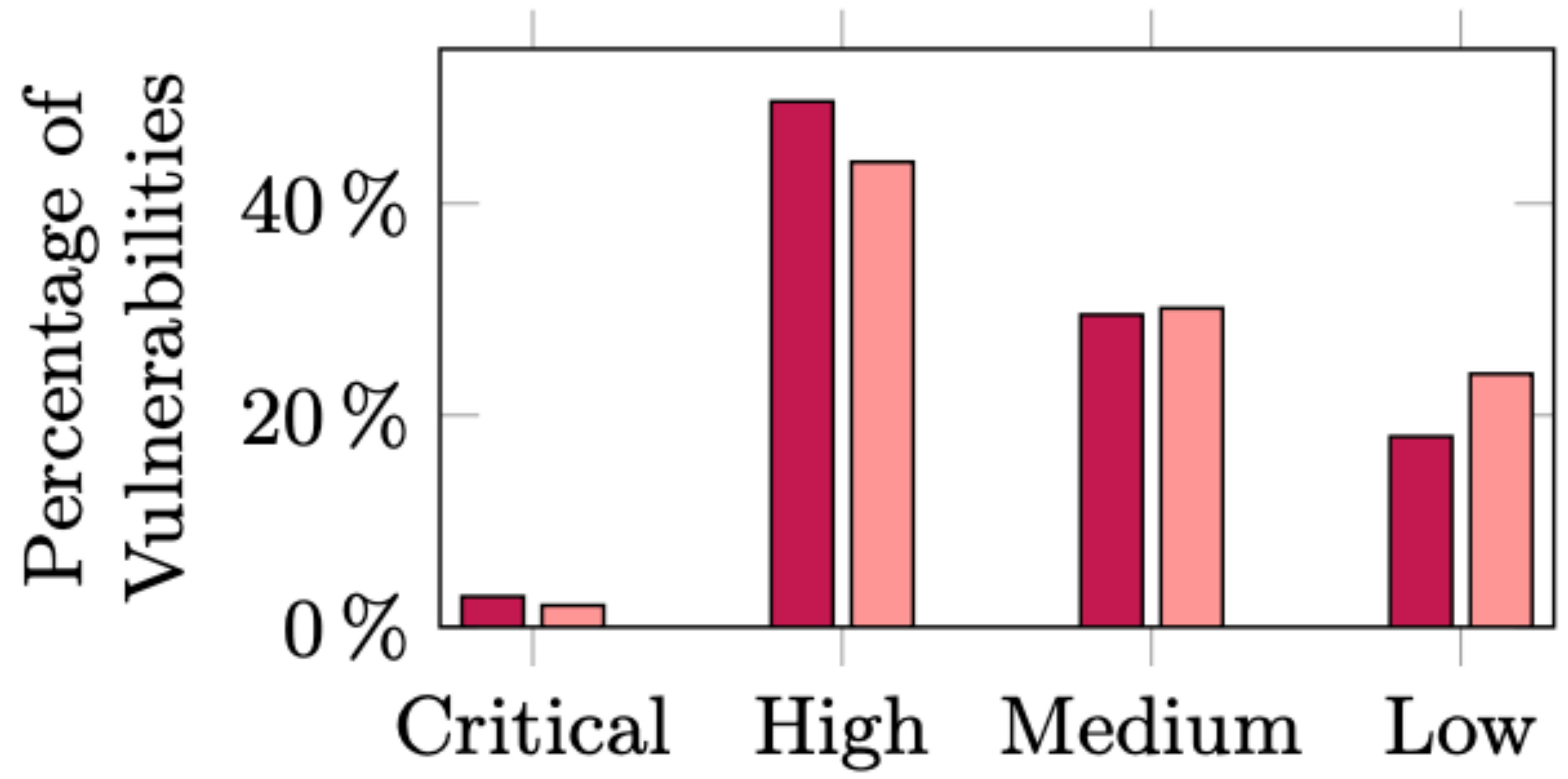
(e) Programming Language

Research Question 3

Additional Results



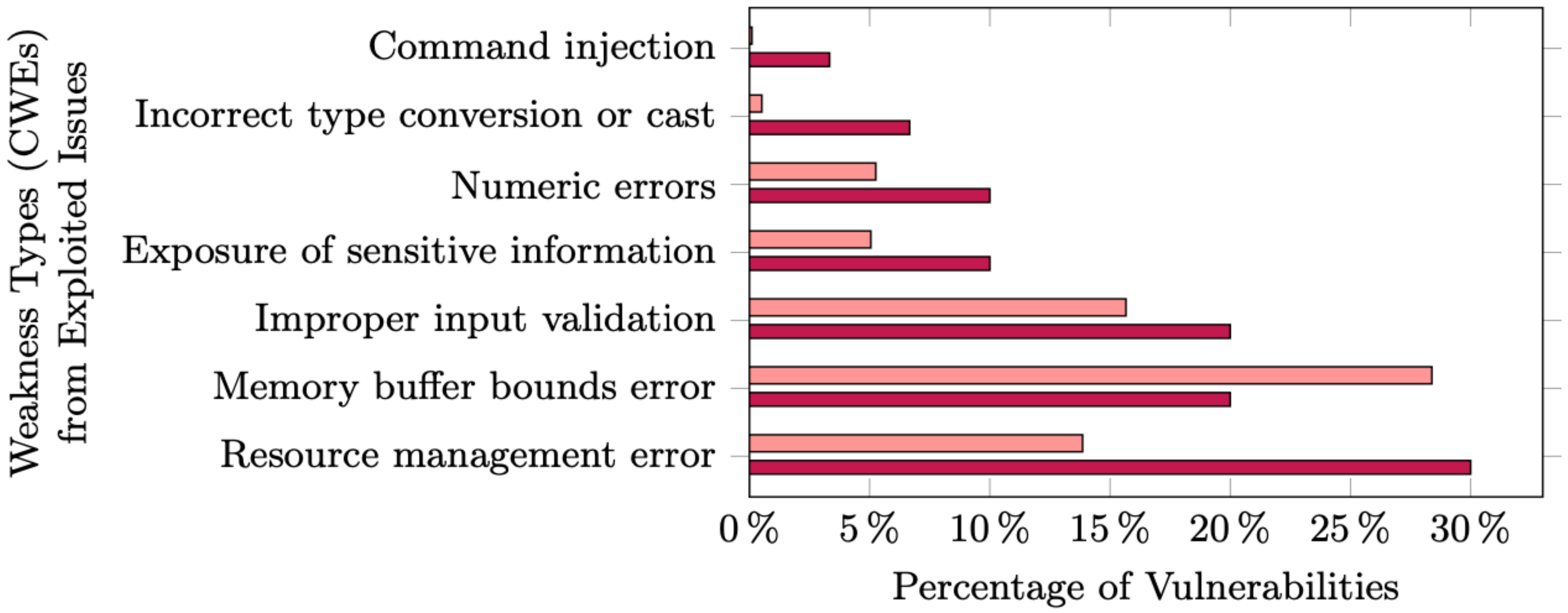
(a) Release Channel



(b) Security Severity

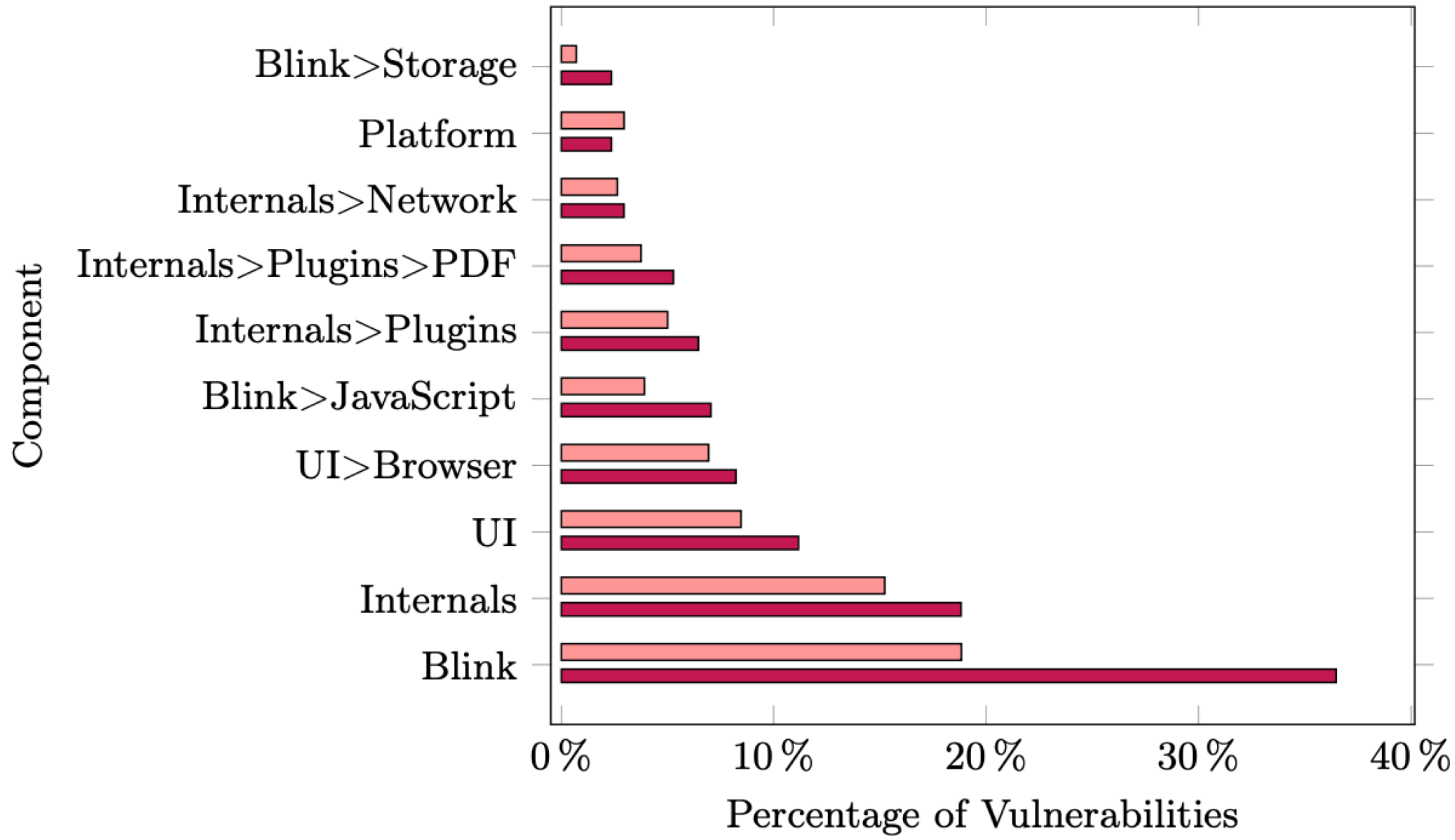
Research Question 3

Additional Results



(c) Weakness Type

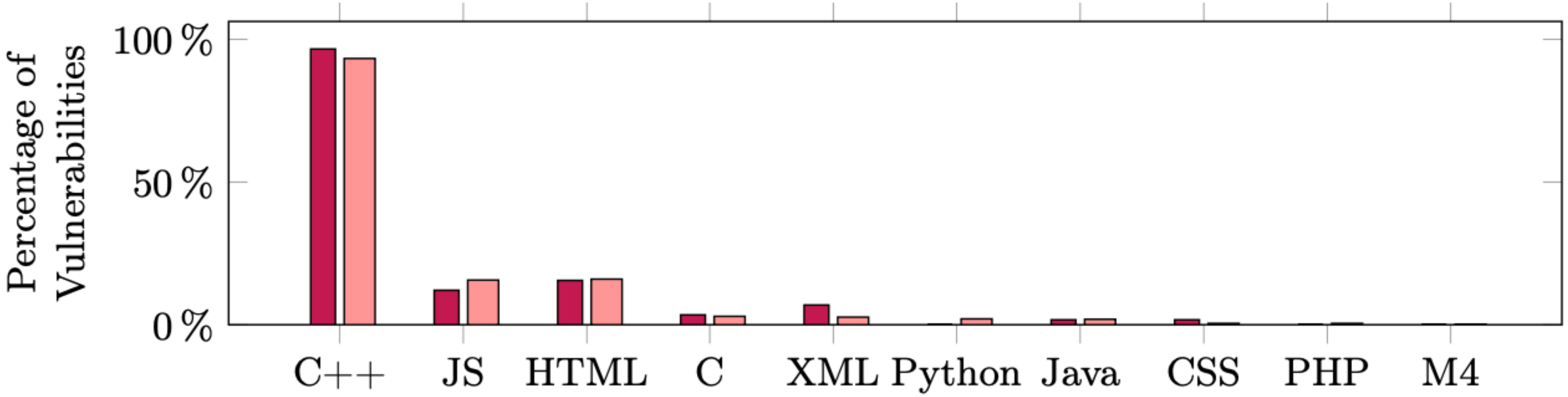
Research Question 3
Additional Results



(d) Component

Research Question 3

Additional Results



(e) Programming Language