UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
UNIVERSITY OF SZEGED
Department of Software Engineering

A Data-Mining Based Study of Security Vulnerability Types and their Mitigation in Different Languages

Gábor Antal, Balázs Mosolygó, Norbert Vándor, Péter Hegedűs

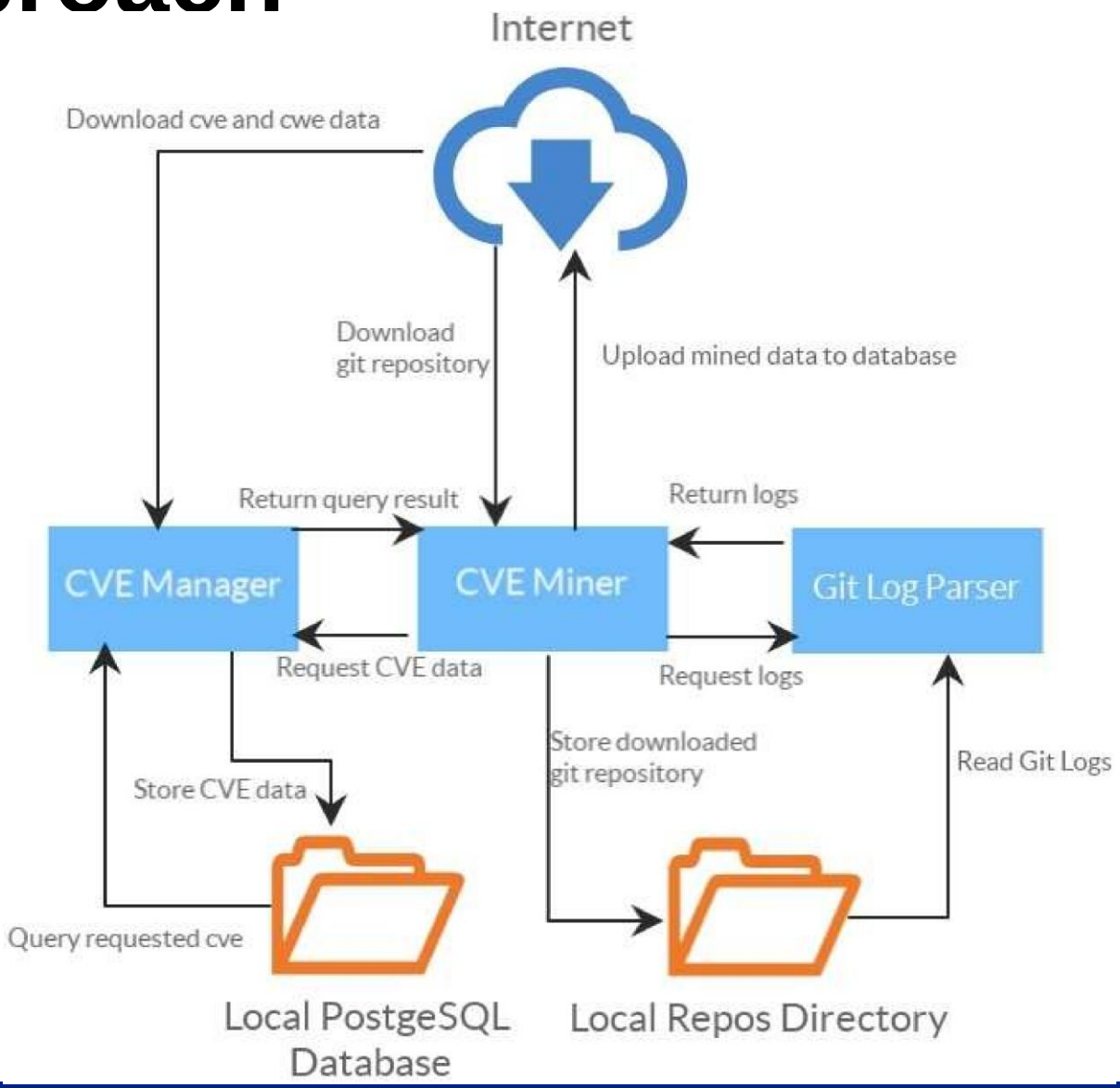
Introduction

- Importance of security and response time
  - More users -> higher potential for abuse
- CVE vs. CWE
- The chosen languages
  - Scheme? BitBake?
- Our goals:
  - Find out if there are noticeable patterns in activity within different programming languages when it comes to finding and fixing vulnerabilities
  - Check if more severe issues get fixed quicker
  - Show the most prevalent weaknesses (CWE) of the languages

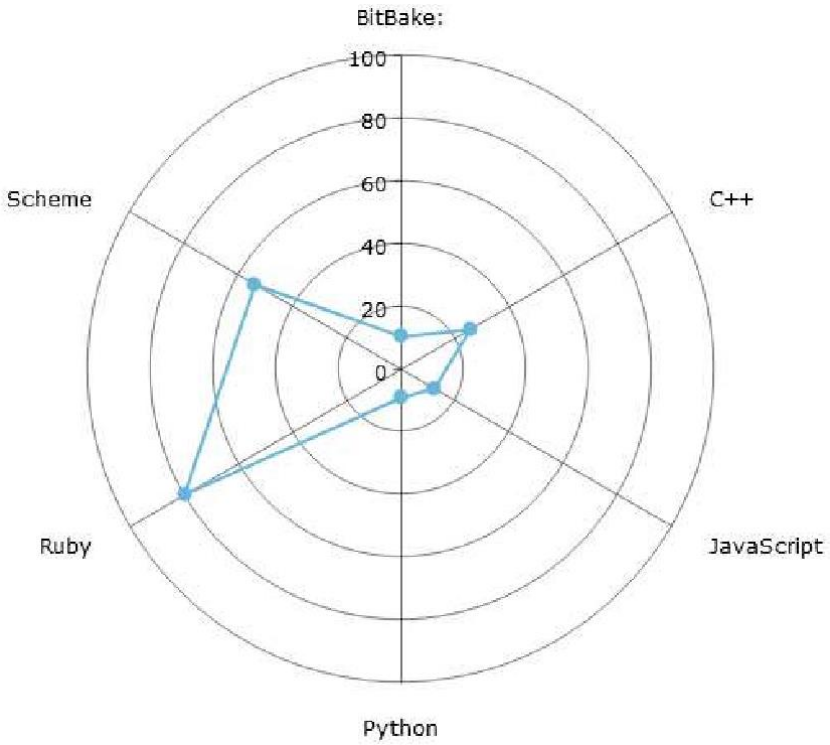
Approach

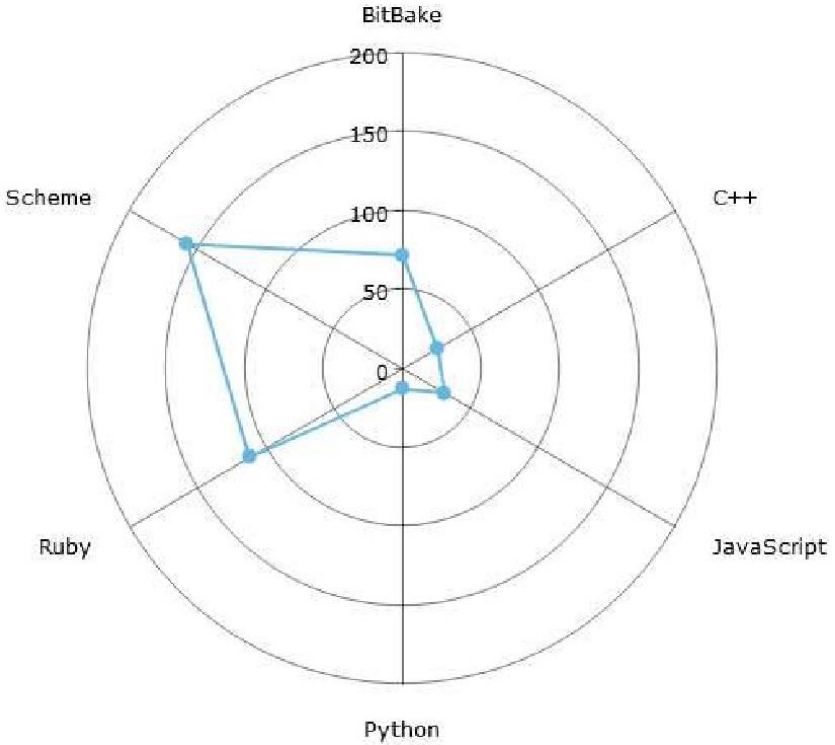
1. Download CVE Data
   - CVE data gets downloaded from MITRE and stored locally
2. Get Git Logs of the project
3. Look for CVE mentions in the logs
4. Verify the existence of the found CVEs and create results

Approach


Internet
Download cve and cwe data
Download
git repository
Upload mined data to database
Return query result
Return logs
CVE Manager
CVE Miner
Git Log Parser
Request CVE data
Request logs
Store downloaded
git repository
Read Git Logs
Store CVE data
Query requested cve
Local PostgeSQL
Database
Local Repos Directory

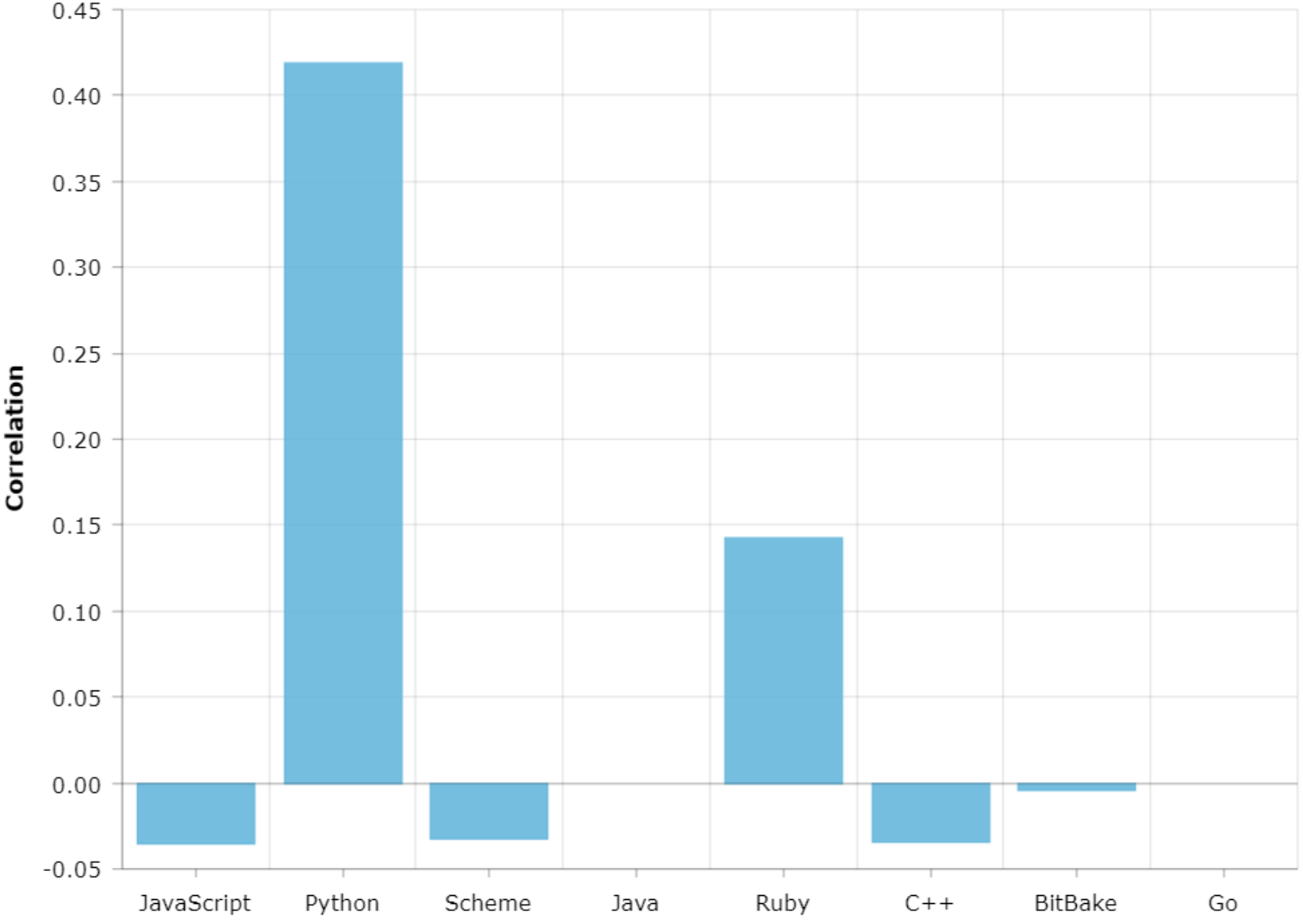
Results





Time elapsed between first mention versus time elapsed since publication

Results


Correlation
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
-0.05
JavaScript
Python
Scheme
Java
Ruby
C++
BitBake
Go

Results

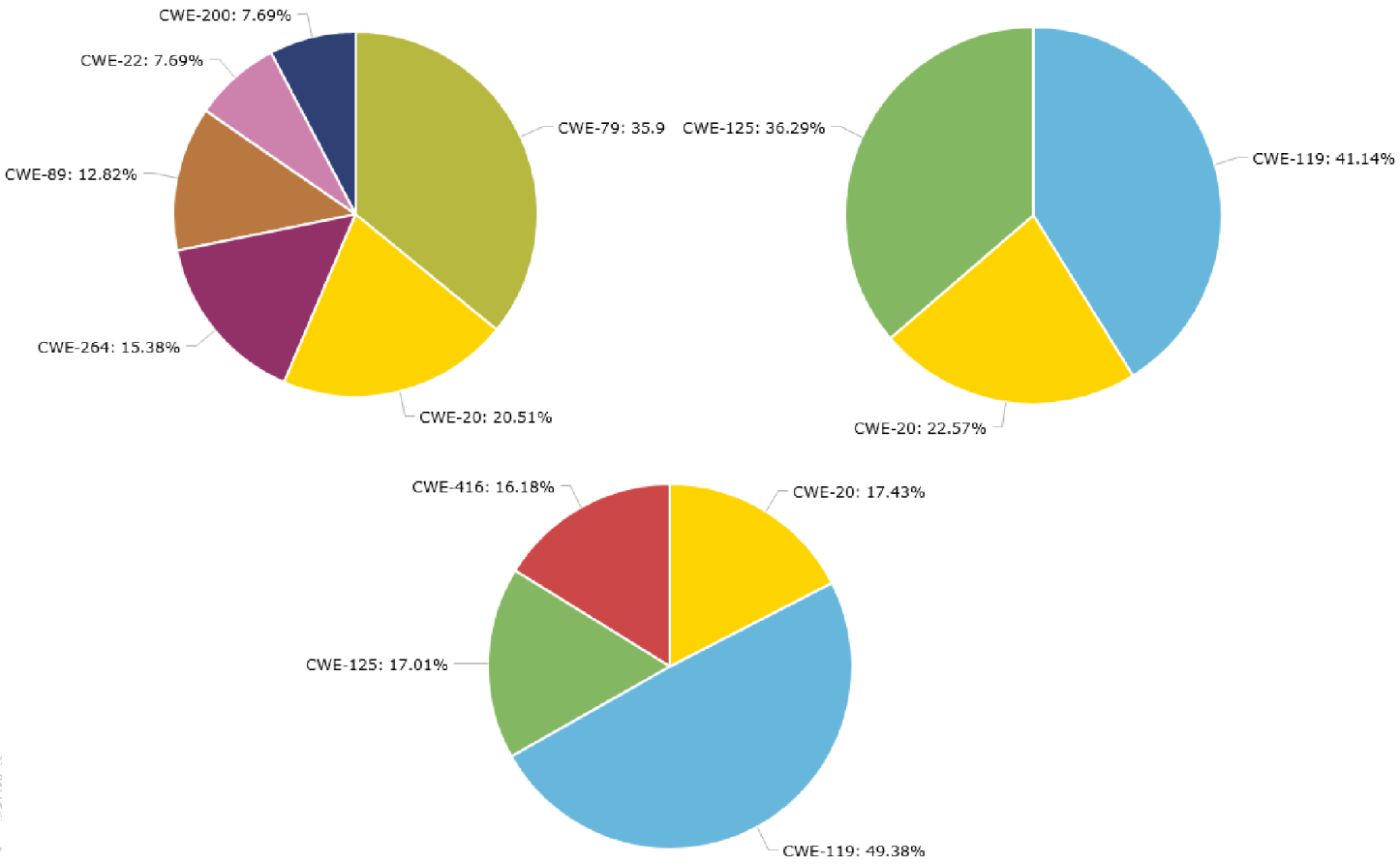
Language	CWE Group	Percentage
BitBake	CWE-119	21.40%
	CWE-20	11.74%
	CWE-125	18.87%
C	CWE-400	25.00%
	CWE-125	25.00%
	CWE-20	50.00%
C++	CWE-119	92.39%
	CWE-200	5.71%
Java	CWE-200	15.00%
	CWE-502	45.00%
	CWE-20	20.00%
JavaScript	CWE-119	6.45%
	NVD-CWE-Other	5.38%
	CWE-20	13.98%
	CWE-400	15.05%
	CWE-200	12.90%
	CWE-79	7.53%

Language	CWE Group	Percentage
Go	CWE-400	25.00%
Python	CWE-200	9.68%
	CWE-79	16.13%
	CWE-601	9.68%
	CWE-185	6.45%
	CWE-20	16.13%
	CWE-89	9.68%
Ruby	CWE-79	26.92%
	CWE-20	15.38%
	CWE-264	11.54%
	CWE-89	9.62%
	CWE-22	5.77%
	CWE-200	5.77%
Scheme	CWE-20	8.29%
	CWE-119	23.49%
	CWE-125	8.09%
	CWE-416	7.70%

Results

- Most common CWEs:
  - Improper restriction of operations within the bounds of a memory buffer (CWE-119)
  - Improper input validation (CWE-20)
  - Out-of-bounds read (CWE-125)
  - Uncontrolled resource consumption (CWE-400)
- Relation between CWE and common language usage

Results



Ruby, BitBake and Scheme represantion

Conclusion

- Languages have typical weaknesses,
  - C++: Improper restriction of operations (CWE-119)
  - Go: Uncontrolled resource consumption (CWE-400)
- Knowing these can be useful for both inexperienced developers and companies
- CVE fix times have a low correlation with their severity
- Issues might reappear later causing worse average fix results