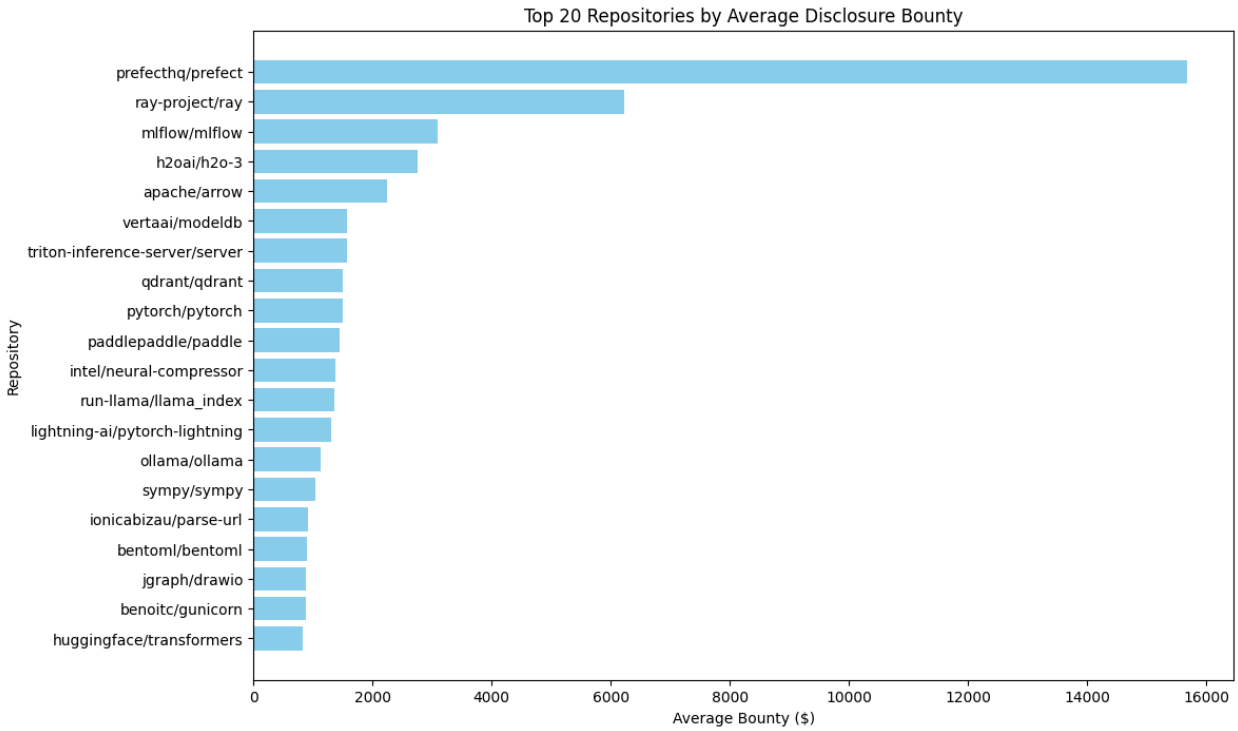
Higher Level Analysis:



Overall Success Rate:

The vast majority of repositories have a 100% success rate for bounty awards. Out of 214 repositories listed, 188 (about 88%) have a 100% success rate.

CWE Specific Analysis:

Most common CWEs:

CWE-79: Cross-site Scripting (XSS) - Stored: 270 occurrences
CWE-79: Cross-site Scripting (XSS) - Reflected: 85 occurrences
CWE-352: Cross-Site Request Forgery (CSRF): 83 occurrences
CWE-284: Improper Access Control: 66 occurrences
CWE-89: SQL Injection: 46 occurrences
CWE-94: Code Injection: 46 occurrences
CWE-29: Path Traversal: '\..\filename': 45 occurrences
CWE-22: Path Traversal: 44 occurrences
CWE-79: Cross-site Scripting (XSS) - Generic: 44 occurrences
CWE-20: Improper Input Validation: 39 occurrences

Repo-wise analysis

While XSS is common, it's not necessarily the most lucrative for bounty hunters. I also analyzed most expensive CWE across all repos and the most expensive CWE for each repo:
1. Most Expensive CWE Overall:
CWE-598 (Use of GET Request Method With Sensitive Query Strings) has the highest average bounty at $4,500.00. This suggests that exposing sensitive information in URLs is considered a critical vulnerability.

2. Bounty Range:
The bounties range from as low as $5.00 to as high as $30,485.00.
Many repositories have an average bounty of $11.00, which might be a default or minimum bounty amount.

3. High-Value CWEs:

CWE-78 (OS Command Injection): Seen in repositories like ray-project/ray with a bounty of $30,485.00.
CWE-94 (Code Injection): Also seen with high bounties, up to $30,485.00 in h2oai/h2o-3.
CWE-400 (Denial of Service): Bounties range from $11.00 to $750.00.

4. Common CWEs:
CWE-79 (Cross-site Scripting - XSS): This appears frequently across many repositories, with bounties ranging from $11.00 to $3,000.00.
CWE-89 (SQL Injection): Also common, with bounties mostly around $11.00, but reaching up to $1,440.00.

5. Repository-specific Observations:

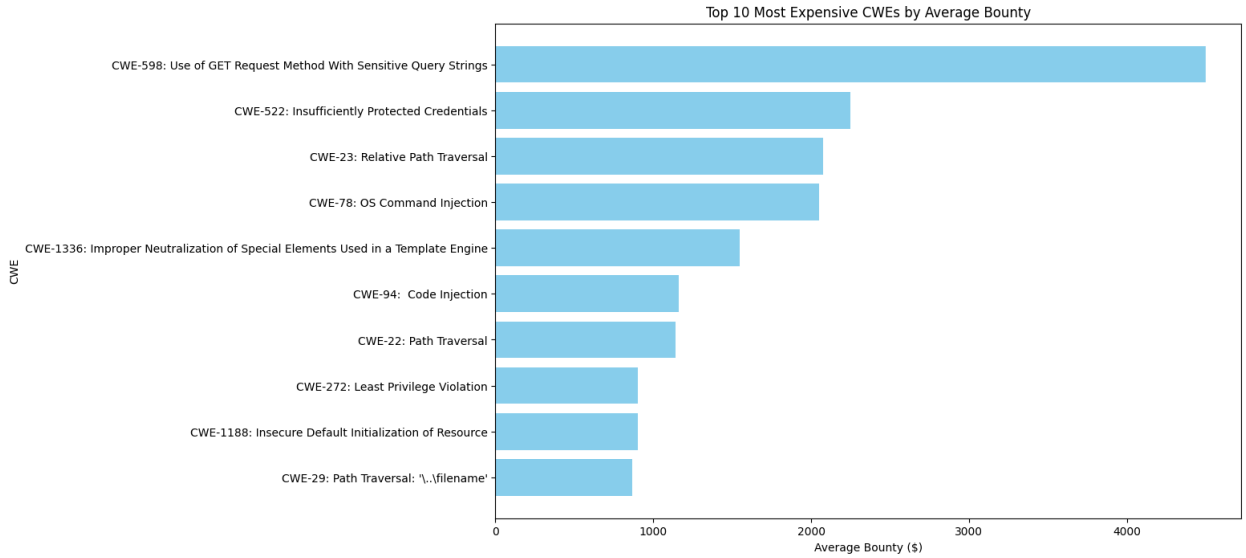
Some repositories like pytorch/pytorch, huggingface/transformers, and gradio-app/gradio have consistently higher bounties.Open-source AI and machine learning projects (like ray-project/ray, h2oai/h2o-3) also tend to have higher bounties/

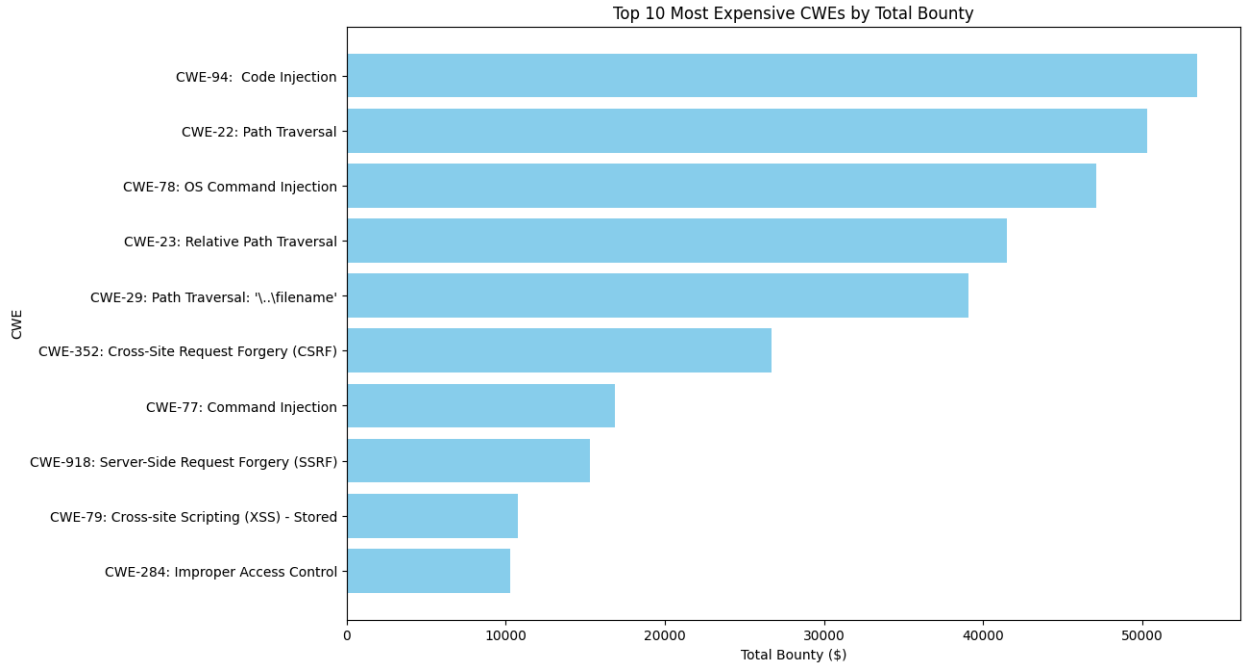
6. Less Common but High-Value CWEs:

CWE-502 (Deserialization of Untrusted Data): Seen with bounties up to $11,665.33 in mlflow/mlflow.
CWE-918 (Server-Side Request Forgery): Bounties range from $11.00 to $915.00.

Across all repos:

Here's a plot of the top 10 most expensive CWEs by average bounty and total bounty. These plots help inform us about what CWEs to focus on:

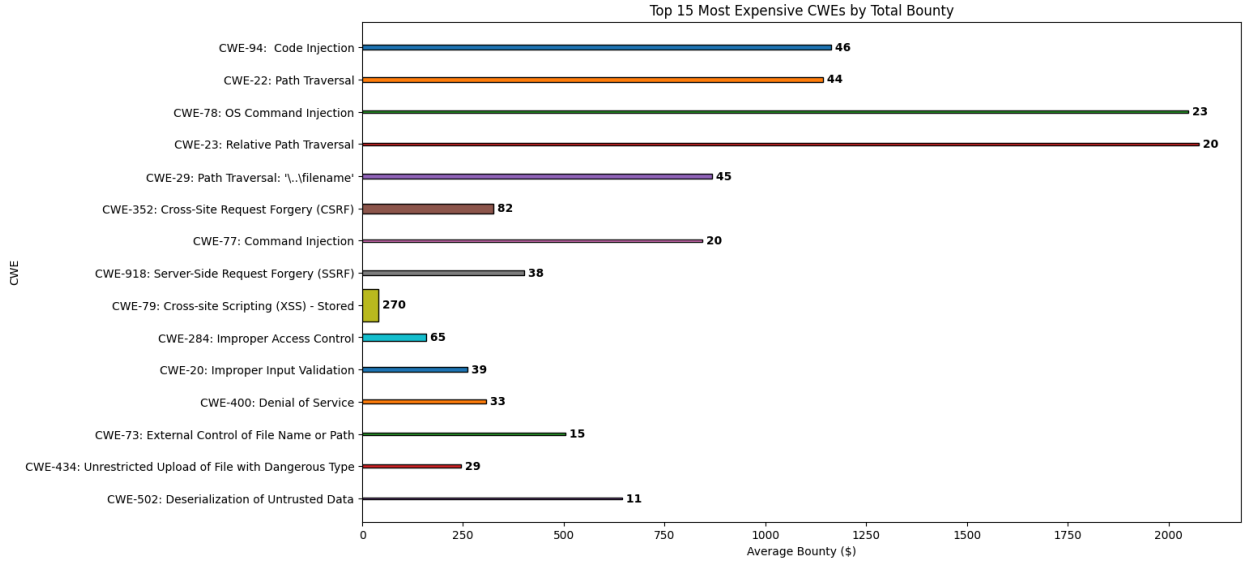




Combining the 2 plots, we get the following plot:

The length of the bar is the average bounty and the width of the bar is the number of total occurrences of this particular CWE across all repos. Therefore, the area of the bar is the total

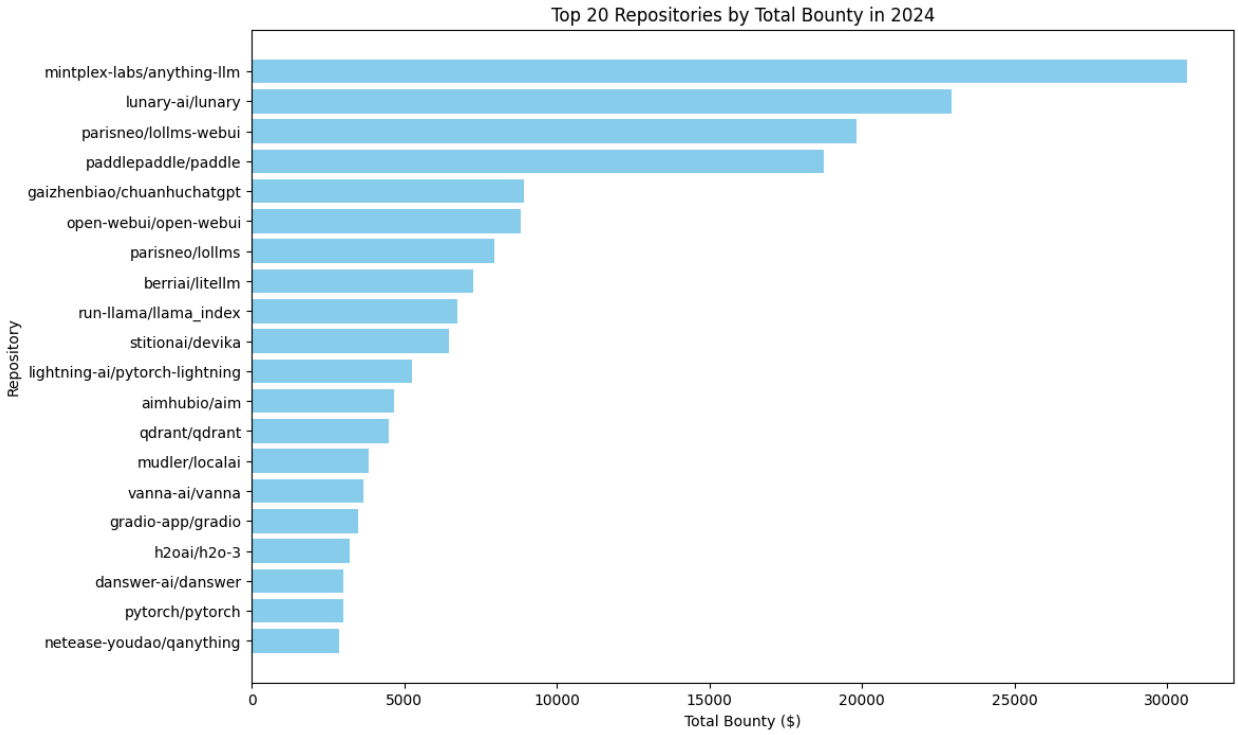
bounty value for this CWE.

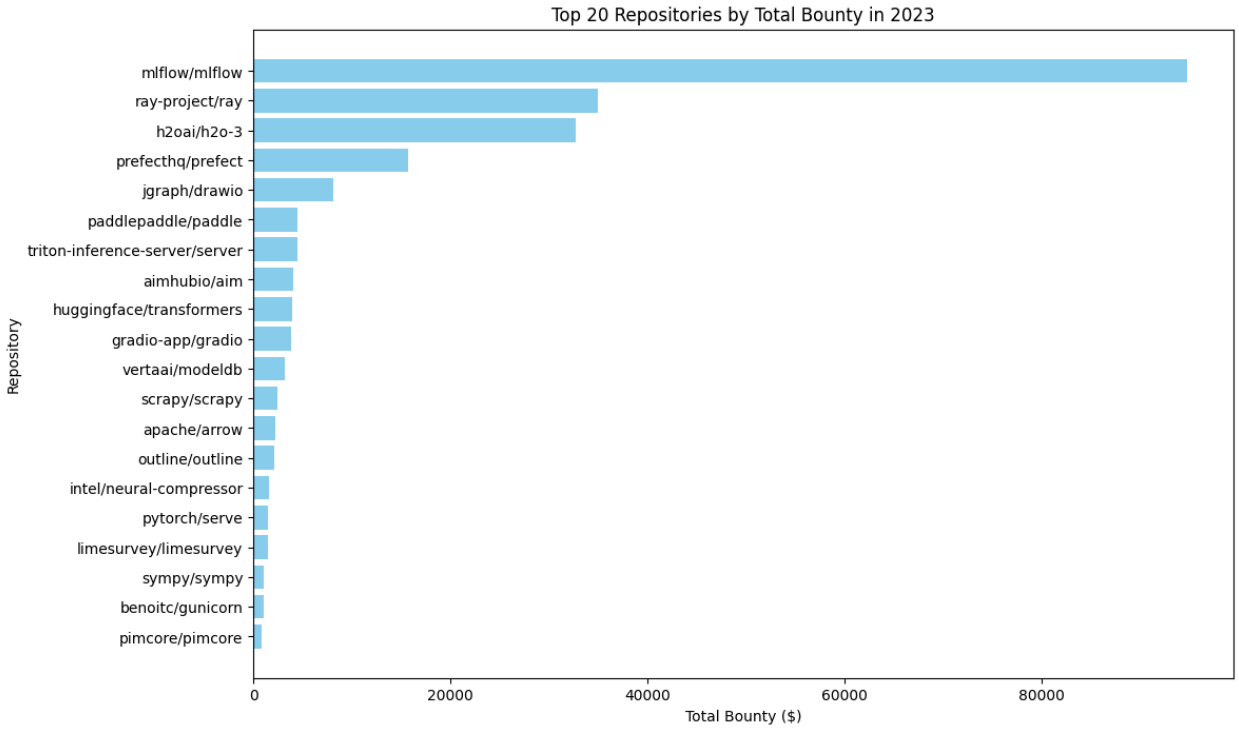


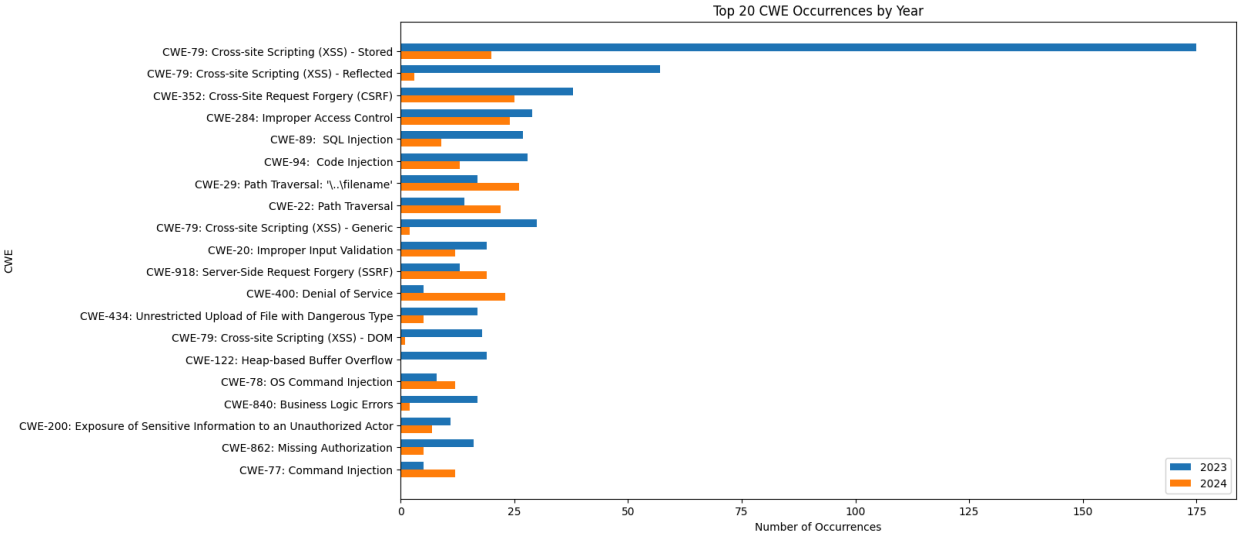
Year comparisons:

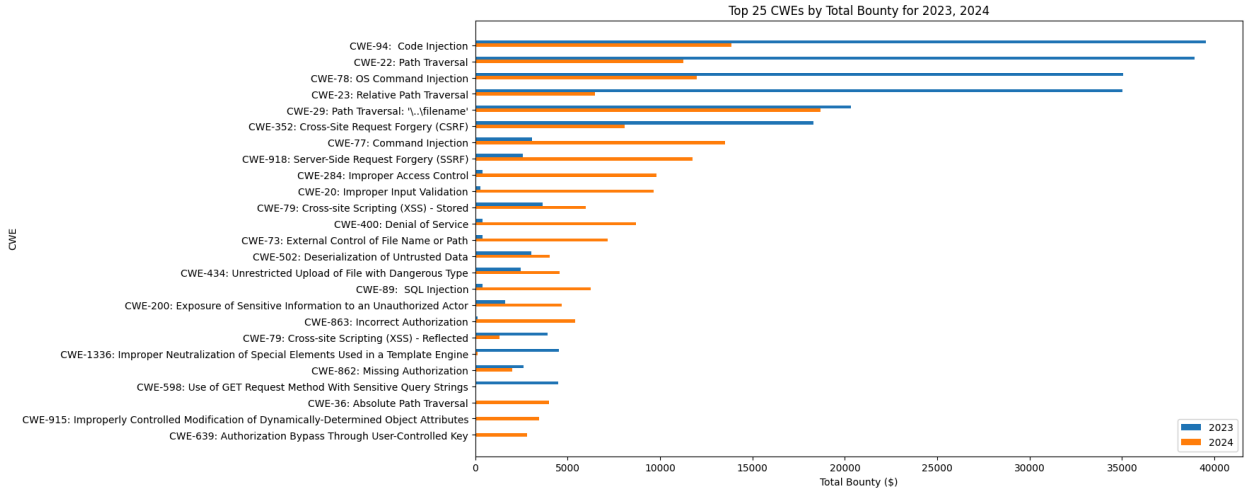
Year comparison plots help us understand how trends change across years.
Note: this will be more informative once we have more data. Right now I'm just using the scraped folder of data which doesn't include a full year for 2024.

We first look at repo level analysis on how total bounty changed in 2024 and 2023. After that we look at a cwe level analysis on frequency and total bounty value in 2024 and 2023:




Top 20 Repositories by Total Bounty in 2023
mlflow/mlflow
ray-project/ray
h2oai/h2o-3
prefecthq/prefect
jgraph/drawio
paddlepaddle/paddle
triton-inference-server/server
aimhubio/aim
huggingface/transformers
gradio-app/gradio
vertaai/modeldb
scrapy/scrapy
apache/arrow
outline/outline
intel/neural-compressor
pytorch/serve
limesurvey/limesurvey
sympy/sympy
benoitc/gunicorn
pimcore/pimcore
Repository
0
20000
40000
60000
80000
Total Bounty ($)


Top 20 CWE Occurrences by Year
CWE-79: Cross-site Scripting (XSS) - Stored
CWE-79: Cross-site Scripting (XSS) - Reflected
CWE-352: Cross-Site Request Forgery (CSRF)
CWE-284: Improper Access Control
CWE-89: SQL Injection
CWE-94: Code Injection
CWE-29: Path Traversal: '\..\filename'
CWE-22: Path Traversal
CWE-79: Cross-site Scripting (XSS) - Generic
CWE-20: Improper Input Validation
CWE-918: Server-Side Request Forgery (SSRF)
CWE-400: Denial of Service
CWE-434: Unrestricted Upload of File with Dangerous Type
CWE-79: Cross-site Scripting (XSS) - DOM
CWE-122: Heap-based Buffer Overflow
CWE-78: OS Command Injection
CWE-840: Business Logic Errors
CWE-200: Exposure of Sensitive Information to an Unauthorized Actor
CWE-862: Missing Authorization
CWE-77: Command Injection
CWE
2023
2024
0
25
50
75
100
125
150
175
Number of Occurrences



N-gram/ Most Common CWE Analysis:

By allowing the model to guess the top 3 CWEs for each repository, you achieved a high accuracy of 73.85%. Overall the model tend to pretty consistent vulnerabilities. Here’a an example output where the percentage following the CWE is the percentage this CWE occurs out of all CWE occurrences for that repo:
Repository: pkp/pkp-lib, Top 3 CWEs: CWE-79 (46.67%), CWE-352 (26.67%), CWE-1241 (6.67%)

If we tweak this approach by adding an additional weighting by average bounty value for that CWE, we get Model Accuracy: 72.43%. For example, when we add a 20% weighting by average bounty value, we now have this result for the example:
Repository: pkp/pkp-lib, Top 3 CWEs: CWE-79 (58.67%), CWE-1241 (34.67%), CWE-352 (30.67%)