Technical Leverage Analysis in the Python Ecosystem

What is Technical Leverage?
A novel metric for measuring dependencies: Ratio between the dependency codes and the original codes in a software package (Massacci and Pashchenko, ICSE’2021).

Technical Leverage = LOC of Dependency code / LOC of Own code

Dataset

21,205 package versions in PyPI

Analysis Procedure
1. Statistical Analysis
2. Mathematical Estimations
3. Simulations

Data in Details

Trend: Small Big

RQ1
How is the Python ecosystem regarding technical leverage and developers’ behavior?

As in Java, Python developers also tend to ship a lot of other people’s code.

RQ2
How does the technical leverage metric change across versions in a package?

If you are highly leveraged, you will stay so.

Probabilities of Getting Safe Package Versions

<table>
<thead>
<tr>
<th>Package Group</th>
<th>Standard Calculation #VulnVersion/#Versions</th>
<th>Our Formula for What Developers Actually Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Not considering downloads</td>
<td>Considering downloads</td>
</tr>
<tr>
<td>No dependencies (TL = 0)</td>
<td>78.86%</td>
<td>89.41%</td>
</tr>
<tr>
<td>Below industry avg. (0&lt;TL&lt;4)</td>
<td>71.89%</td>
<td>81.01%</td>
</tr>
<tr>
<td>Above industry avg. (TL &gt; 4)</td>
<td>68.93%</td>
<td>89.91%</td>
</tr>
</tbody>
</table>

RQ3
How does the technical leverage metric affect the risk of having vulnerabilities in Python ecosystem?

The CHANCE of getting a SAFE PACKAGE VERSION is HIGHER than just reporting the percentage of vulnerable versions.

Future Works
1. How do packages’ security states evolve over time?
2. How to do security MSR research with the available knowledge at a certain point in time?

RANINDYA PARAMITHA
ranindya.paramitha@unitn.it
(1) Università di Trento, Italy; (2) Vrije Universiteit Amsterdam, The Netherlands

FABIO MASSACCI
fabio.massacci@ieee.org