Unmasking Cybersecurity Vulnerabilities in Direct and Transitive OSS Dependencies

Alexis, Sebastian (School: Northwood High School)

Open-source software(OSS) is free and widely used in software development, from video games to systems that run critical infrastructure such as power plants. However, most OSS libraries are built upon a chain of other dependent OSS libraries known as transitive dependencies. Often, software developers may not be aware of whether the OSS they use contains vulnerable, dependent libraries, which could potentially be exploited by cybercriminals. I developed SecureOSS, a multi-modular cybersecurity product that interfaces with the National Vulnerability Database (NVD) to identify vulnerabilities in both direct and transitive OSS dependencies. The workflow starts with the Dependency Parser, which uses a novel version of the Depth First Search Algorithm to traverse the project's dependency tree. As it parses, it generates the Software Bill of Materials(SBOM) in CycloneDX format which catalogs the project's direct and transitive dependencies. It also generates a supplemental report of Community Health Analytics, including Open Source Software (CHAOSS) metrics which enriches the SBOM by offering insights into the activity and responsiveness of OSS projects. For the identified dependencies, the Concurrency Manager efficiently orchestrates parallel queries, leveraging the NVD API Interface to search for Common Vulnerabilities and Exposures(CVE). The Cache Manager stores frequently accessed dependencies, reducing redundant API calls while the Rate Limiter ensures adherence to NVD's request policies. Using NVD’s Common Vulnerability Scoring System(CVSS) scores, the vulnerabilities' impact is found. My product successfully identified significant vulnerabilities, including the Log4j issue (CVE-2021-44228), with a high accuracy rate of 95% and a minimal error rate of 2%