Knowledge Graph Construction for Resilient, Trustworthy, and Secure Software Supply Chains

Tianyi Zhang, Purdue University; Muhao Chen, Xiang Ren, Xiangyu Zhang

Use case description and societal challenge being addressed

Our software supply chain knowledge graph continually monitors and tracks dependencies between software and hardware components and their vulnerabilities reported or discussed in online platforms such as CVE and The Hacker News. We describe two use cases below.

- First, our KG can be used by developers and engineers in IT companies, federal agencies, and the military to secure the software they develop. They can query the KG to check whether any of their third-party software libraries or firmware they use when building their software, including the transitive dependencies, contain any vulnerabilities reported or discussed in the past. If a software dependency is reported to have a vulnerability, they can further query the KG to find which version of the dependency is fixed and whether there are any alternative dependencies.

- Second, our KG can be used by IT managers in a company, federal agency, or military to decide whether a software product is safe to purchase. According to the President's Executive Order (EO) 14028 on Improving the Nation's Cybersecurity in 2021, all software vendors need to provide a comprehensive list of the software components that comprise their products (known as Software Bill of Materials, or SBOM). It is time-consuming for IT managers to manually vet all components in an SBOM, especially given the complex transitive dependencies between software components. IT managers can instead query the KG to check whether any component in an SBOM and its transitive dependencies have known vulnerabilities.

Software supply chains have emerged as an indispensable asset across a wide array of industries. According to the 2024 OSSRA report, 96% of software applications use open-source software. The scope and scale of software supply chains open up many attack surfaces for adversaries. A risk in upstream software components can quickly propagate to a wide range of downstream software systems. Our KG can increase the transparency of software supply chains and prevent lurking security issues in upstream dependencies from slipping into downstream software applications.

Knowledge graph source datasets

- The Libraries.io dataset contains open-source software libraries and applications from 33 package managers and 3 source code hosting platforms.
- CVE List includes publicly disclosed cybersecurity vulnerabilities.
- The Hacker News includes the latest vulnerabilities from many sources.

We expect our KG to have 1.7M software entities, 53K hardware entities, 16.5M version entities, 503 vulnerability entities, 4.9M edges between vulnerability entities and software/hardware entities, and 190M edges between software/hardware entities.

User queries / competency queries for the use case

- Does the Apache Struts framework have any known vulnerabilities?
- What software libraries and packages are Apache Struts built upon?
- Does Apache Struts depend on any software library or package, directly or transitively, that has a known vulnerability?
- Which version of Apache Struts no longer has the Remote Code Execution vulnerability?