**Caracterización de la vulnerabilidad de redes de dependencias de paquetes**

El desarrollo *open source* ha proporcionado una gran cantidad y diversidad de librerías que se organizan en índices, normalmente publicados en repositorios oficiales configurados por defecto en las herramientas de instalación y gestión de paquetes de cada lenguaje. Algunos ejemplos son PyPI (Python Package Index) para Python, NPM (Node Package Manager) para JavaScript (Node.js), PEAR (PHP Extension and Application Repository), CPAN (Comprehensive Perl Archive Network) o Maven Central para Java. El tamaño típico de estos índices es del orden de cientos de miles de referencias.

La utilización de librerías de estos repositorios para disminuir los tiempos de desarrollo y costes es prácticamente universal, en todos los lenguajes y tipos de proyecto software. Sin embargo, su empleo introduce riesgos como la exposición a bugs y modificaciones malintencionadas, los derivados del mantenimiento y actualización del software y, en última instancia, los propios de la dependencia funcional de una tercera parte. Estos riesgos pueden ser difíciles de apreciar en toda su dimensión por los desarrolladores, que sólo importan explícitamente una pequeña parte de las librerías que se emplean en cada proyecto. Debido a la transitividad de las dependencias, un único defecto o modificación puede tener efectos extensos y difíciles de predecir en el ecosistema software.

En este trabajo se construirá un modelo de red de dependencias de paquetes para el análisis de su vulnerabilidad frente a fallos y ataques. El estudio de este tipo de modelos es reciente y su comprensión teórica limitada [1] [2]. Se desarrollará un conjunto de herramientas para la extracción de datos y construcción y evaluación del modelo y los resultados teóricos se pondrán a prueba con alguno de los repositorios reales de paquetes de uso común.

[1] Kikas, R., Gousios, G., Dumas, M., & Pfahl, D. (2017, May). Structure and evolution of package dependency networks. In 2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR) (pp. 102-112). IEEE.

[2] Boldi, P. (2019, December). How Network Analysis Can Improve the Reliability of Modern Software Ecosystems. In *2019 IEEE First International Conference on Cognitive Machine Intelligence (CogMI)* (pp. 168-172). IEEE.