Using machine learning to detect software vulnerabilities

July 24, 2018 by Ingrid Fadelli, Tech Xplore

- Home
- Security
- July 24, 2018

Credit: Markus Spiske on Unsplash

A team of researchers from R&D company Draper and Boston University developed a new large-scale vulnerability detection system using machine learning algorithms, which could help to discover software vulnerabilities faster and more efficiently.

Hackers and malicious users are constantly coming up with new ways to compromise IT systems and applications, typically by exploiting software security vulnerabilities. Software vulnerabilities are small errors made by the programmers who developed a system that can propagate quickly, especially through open-source software or through code reuse and adaptation.

Every year, thousands of these vulnerabilities are publicly reported to the Common Vulnerabilities and Exposures database (CVE), while many others are spotted and patched internally by developers. If they are not adequately addressed, these vulnerabilities can be exploited by attackers, often with devastating effects, as proved in many recent high-profile exploits, such as the Heartbleed bug and the WannaCry ramsomware cryptoworm.

Generally, existing tools to analyze programs can only detect a limited number of potential errors, which are based on predefined rules. However, the widespread use of open-source repositories has opened new possibilities for the development of techniques that could reveal code vulnerability patterns.

The researchers from Draper and Boston have developed a new vulnerability detection tool that uses machine learning for automated detection of vulnerabilities in C/C++ source code, which has already showed promising results.

The team compiled a large dataset with millions of open-source functions and labeled it using three static (pre-runtime) analysis tools, namely Clang, Cppcheck and Flawfinder, which are designed to identify potential exploits. Their dataset included millions of function-level examples of C and C++ code drawn from the SATEIV Juliet Test Suite, Debian Linux distribution, and public Git repositories on GitHub.

"Using these datasets, we developed a fast and scalable vulnerability detection tool based on deep feature representation learning that directly interprets lexed source code," the researchers wrote in their paper.

As programming languages are in some ways similar to human languages, the researchers designed a vulnerability detection technique that uses natural language processing (NLP), an AI strategy that allows computers to understand and interpret human language.

"We leverage feature-extraction approaches similar to those used for sentence sentiment classification with convolutional neural networks (CNNs) and recurrent neural networks (RNNs) for function-level source vulnerability classification," the researchers explained in their paper.

They combined NLP with random forest (RM); a powerful algorithm that creates an ensemble of decision trees from randomly selected subsets of the training dataset and then merges them together, generally achieving more accurate predictions.

The researchers tested their tool on both real software packages and the NIST STATE IV benchmark dataset.

"Our results demonstrate that deep feature representation learning on source code is a promising approach for automated software vulnerability detection," they wrote. "We applied a variety of ML techniques inspired by classification problems in the natural language domain, fine-tuned them for our application, and achieved the best overall results using features learned via convolutional neural network and classified with an ensemble tree algorithm."

So far, their work has focused on C/C++ code, but their method could also be applied to any other programming language. They specifically chose to create a custom C/C++ lexer as this would produce a simple and generic representation of function source code, which is ideal for machine learning training.

Explore further: What are software vulnerabilities, and why are there so many of them?

More information: Automated Vulnerability Detection in Source Code Using Deep Representation Learning. arXiv:1807.04320v1 [cs.LG]. arxiv.org/abs/1807.04320

Abstract
Increasing numbers of software vulnerabilities are discovered every year whether they are reported publicly or discovered internally in proprietary code. These vulnerabilities can pose serious risk of exploit and result in system compromise, information leaks, or denial of service. We leveraged the wealth of C and C++ open-source code available to develop a large-scale function-level vulnerability detection system using machine learning. To supplement existing labeled vulnerability datasets, we compiled a vast dataset of millions of open-source functions and labeled it with carefully-selected findings from three different static analyzers that indicate potential exploits. Using these datasets, we developed a fast and scalable vulnerability detection tool based on deep feature representation learning that directly interprets lexed source code. We evaluated our tool on code from both real software packages and the NIST SATE IV benchmark dataset. Our results demonstrate that deep feature representation learning on source code is a promising approach for automated software vulnerability detection.