Using Machine Learning To Statically Analyze JavaScript for Cross-Site Scripting Vulnerabilities

Namikas, Timothy (School: Episcopal School of Baton Rouge)

Cross-site scripting (XSS) attacks are one of the most common web vulnerabilities but are difficult to track and prevent. Websites have a very large XSS attack surface area, with every user input on a website leaving potential spaces for a vulnerability. Currently, the most effective method for identification is taint tracking, a form of dynamic analysis. Its high precision in detection comes at the cost of resource consumption and time constraints that make it impractical for many applications. Using a database of Javascript functions scraped from the web and labeled as vulnerable or safe, I compared two machine learning models, a Random Forest Decision Tree and a Deep Neural Network (DNN). The models were trained on data I processed using two different hash functions, a standard hash and a Term-Frequency Inverse Document Frequency (TFIDF). I found that the DNN trained on the TFIDF hash function maintained the highest accuracy at a high recall. Combining the DNN and taint tracking into one system would reduce the number of functions that taint tracking needs to analyze by 242.2x, at a recall of 99%. The vast reduction in resource overhead and analysis time makes it possible to apply in-browser while maintaining a reasonably high detection rate. Further, I performed LIME (Local Interpretable Model-agnostic Explanations) interpretability analysis to gain novel insight into static analysis techniques. This analysis can begin to uncover the details of how machine learning models make their predictions, which would improve our understanding of the dataset and characteristics of XSS attacks.

Awards Won: