Even thought the problems of buffer overruns have existed since the C language was created the vulnerability is still found in code written today. The authors of this article aim to curb some of those errors by introducing an automated static analysis tool to search for overruns in source code. Their algorithm is not sophisticated enough to avoid false positives, but it has caught errors where a security expert failed to. Another large benefit of this tool is that it can scale to large repositories of code. What would have taken hours for a reviewer to scan with their eye can be done in moments. The authors are quick to add as well that their tool does not catch every overrun but should be a tool in the hands of those reviewing the code’s security. I care about this research because my proposed topic for the class is using machine learning to automate detection of bad input parameters. This article gives me direction to look more into static analysis and integer range analysis as tools similar to machine learning.

Wagner, David, et al. "A First Step Towards Automated Detection of Buffer Overrun Vulnerabilities." *NDSS*. 2000.