Discovering Vulnerable Functions: A Code Similarity Based Approach

by

aditya.chandran , Lokesh Jain, Sanjay Rawat, Kannan Srinathan

in

4th International Symposium Security in Computing and Communications 2016

Report No: IIIT/TR/2016/-1

Centre for Security, Theory and Algorithms
International Institute of Information Technology
Hyderabad - 500 032, INDIA
September 2016

Discovering Vulnerable Functions: A Code Similarity Based Approach

Aditya Chandran, Lokesh Jain , Sanjay Rawat, Kannan Srinathan
Chapter

Security in Computing and Communications

Volume 625 of the series Communications in Computer and Information Science pp 390-402
Date: 17 September 2016

Abstract

This paper extends recent work on vulnerability extrapolation. A surge in vulnerability exploits against old and new softwares, urges the importance of detection of vulnerabilities and possible attacks prior to the attacker. How sophisticated an exploit may be, an underlying prerequisite remains to be the presence of at least one memory corruption bug, serving as entry point for the exploit. Therefore several rigorous software testing techniques are borrowed to detect and eliminate software bugs as early as possible. Code similarity based bug detection is one of such techniques, which, in the parlance of software security, is also termed as vulnerability extrapolation. In this paper, we present a source code similarity based bug identification technique by considering code features that are relevant for security related bugs. Our technique works by enriching (augmenting) abstract syntax trees (ASTs) of functions by considering security relevant properties of the code. We show the effectiveness of the augmented AST based similarity approach over existing methods by evaluating proposed method on real-world applications.
Keywords
Software vulnerability Abstract syntax tree Vulnerability extrapolation Code similarity