**A Hybrid Approach to Combating High-Level Semantic Attacks**

**By Stu Steiner**

**Ph.D. Proposal Defense**

**Abstract:**

Over the past few years, there has been a rapid increase in interactive web sites that offer a wide range of services including shopping carts, e-commerce transactions, social media interactions and much more. A majority of these websites contain a database backend where user information, and product information is stored and then retrieved when needed.

The Open Web Application Security Project (OWASP) maintains a list of the top ten vulnerabilities. This top ten list coincides with OWASP's mission to make software security visible to all, so individuals and organizations can make intelligent decisions about software security.

One challenge inherent with software security is how to combat injection vulnerabilities, such as SQL injection. Injection vulnerabilities occur when untrusted data is passed to the database interpreter without proper sanitization. Passing untrusted data to the interpreter can allow for exploits in the web application, such as revealing sensitive user information, data corruption, denial of access or complete host takeover. Code that properly sanitizes the untrusted data can easily combat injection vulnerabilities; however, an inexperienced developer may not understand or have knowledge of this concept or how to implement security measures.

Current techniques to combat injection vulnerabilities are typically classified into one of three categories, static analysis, dynamic analysis or a combination of both static and dynamic analysis. Static analysis involves verifying that the untrusted data is safe before it is passed to the interpreter. This is usually accomplished with input validation via string tokenization, control-flow graphs and many other techniques. Dynamic analysis, such as dynamic information flow tracking (DIFT), involves executing the interpreter with the untrusted data in a controlled environment and then examining the results of the execution. Static analysis is often ineffective because attackers are continually discovering new attack vectors that are able to avoid the checks static analysis provides. Dynamic analysis is also often ineffective because the rigorous overhead can ultimately slow the system in unacceptable ways. There is limited research on combining both dynamic and static analysis techniques. This dissertation proposes the development of a hybrid approach that combines the best aspects of static and dynamic analysis while avoiding their individual shortcomings. This is accomplished by summarizing the static and dynamic analysis schemes proposed in recent years; introducing a hybrid approach, combining static analysis and dynamic analysis in a unique way limiting interaction from the developer; and, proposing new research to implement, evaluate and improve the detection and avoidance of injection vulnerabilities.

A goal of this project is to formalize the definition of injection vulnerabilities. In addition this dissertation discusses the formal model of injection vulnerabilities and how these vulnerabilities are handled by the interpreter and the hardware, including a policy-based architectural mapping framework. Finally, the remaining work of the dissertation is proposed.

**Tuesday, September 17, 2013**

**1:00 p.m. JEB 326**

**Major Professor: Dr. Jim Alves-Foss**



**Department of Computer Science**