WebHoneyBug

Active defense framework for web applications

Abstract

Web is a very agile technology by its nature and every day we see new platforms and technologies are emerging to make it more interactive. Most web applications are built in a very short period of time because companies prefer to keep up with the new technologies and frameworks and constantly change their web applications over short period of time. This makes the web applications more prone to vulnerabilities. In this proposal, we present WebHoneyBug, an active defense framework, for web applications to make web applications more resilient in the face of web attackers. The aim of this framework is to enable web applications to attribute attackers and deceive them in order to reduce the risk of zero-day vulnerabilities in the web application being exploited by them.

WebHoneyBug achieves these goals by making a shadow copy of the web application and try to modify this shadow by implanting crafted vulnerabilities. In addition, the shadow server uses a shadow database. The structure of this database is similar to the original database but the data is different and not real. Normal user functions will be maintained at all times. WebBugHunter detects attackers by using signature and application behavior based on static analysis. It redirects attack traffic to this shadow server. In addition, all transactions to the shadow instance of the web application will be monitored and in case of observing leakage of any “fake information”, the transactions will be explored to check whether this occurred as the result of exploiting an implanted vulnerability or it’s because of a zero day vulnerability. In case of finding a new zero day vulnerability, a new rule will be generated to prevent that vulnerability to be exploited in the future. In this way, WebHoneyBug will patch the original web application on the fly by using attacker’s knowledge and expertise.