Analyzing Information Flow in JavaScript-based Browser Extensions

Mohan Dhawan and Vinod Ganapathy

Department of Computer Science, Rutgers University

Abstract. JavaScript-based browser extensions (JSEs) enhance the core functionality of the web browser by improving its look and feel, and are widely available for commodity browsers. To enable a rich set of functionalities, the browser typically executes JSEs with elevated privileges. For example, unlike JavaScript code in a web application, JavaScript code in a JSE is not constrained by the same-origin policy. Malicious JSEs can misuse these privileges to compromise confidentiality and integrity, *e.g.,* by stealing sensitive information, such as cookies and saved passwords, or executing arbitrary code on the host system. Even if a JSE is not overtly malicious, vulnerabilities in the JSE and the browser may allow a remote attacker to compromise browser security.

We present Sabre (Security Architecture for Browser Extensions), a system that uses JavaScript-level information-flow to analyze JSEs. Sabre associates a label with each in-memory JavaScript object in the browser, which determines whether the object contains sensitive information. Sabre raises an alert if an object containing sensitive information is accessed in an unsafe way, *e.g.,* if a JSE attempts to send the object over the network or write it to a file. We implemented Sabre by modifying the JavaScript interpreter in Firefox and evaluated it using both malicious JSEs as well as benign ones that contained exploitable vulnerabilities. Our experiments show that Sabre can precisely identify information flow violations.

Key words: JavaScript, browser extensions, information flow tracking.