

# Analyzing Information Flow in JavaScript-based Browser Extensions

Mohan Dhawan and Vinod Ganapathy



# Browser extensions

- Available for all major browsers
  - Plugins, Browser-helper Objects, Add-ons.
- Our focus: JavaScript-based Extensions (JSEs)
  - GreaseMonkey, Firebug, NoScript
- Popular as “add-ons” for Firefox
  - Other browsers also allow JS in extensions.

# JSEs affect browser security

- Inadequate sandboxing of JavaScript in a JSE
  - Not constrained by the Same-Origin Principle
    - JSEs can access cookies, browsing history, location bar
  - Browsers offer APIs that extend JS functionality
    - XPCOM offers access to file system and network
- Browser and JSE vulnerabilities
  - Malicious websites can misuse privileges of JSE to violate confidentiality and integrity

# Inadequate sandboxing

- Malicious JSEs can misuse privileges
- Code snippet from FFsniFF
  - Emails passwords in form fields to attacker

```
function do_sniff() {  
  var hesla = window.content.document.getElementsByTagName("input");  
  data = "";  
  for (var i = 0; i < hesla.length; i++) {  
    if (hesla[i].value != "") {  
      ...  
      data += hesla[i].type + ":" + hesla[i].name + ":" + hesla[i].value + "\n";  
      ...  
    }  
  }  
  // the rest of the code sends 'data' via an email message.  
}
```

# Browser and JSE vulnerabilities

- GreaseMonkey/Firefox vulnerabilities
- GreaseMonkey allows user-defined JavaScript to execute on each loaded webpage
  - Add GM API functions to `window` object
  - Allows user-scripts to access GM APIs
  - Remove GM APIs from `window` before `onload`
- Vulnerability in Firefox allowed malicious JavaScript in a webpage to access GM API

# GreaseMonkey/Firefox vulnerability

```
1. <script type="text/javascript">
2. window._GM_xmlHttpRequest = null;
3. function trapGM(...) {
4.     window._GM_xmlHttpRequest = window.GM_xmlHttpRequest;
5.     ...
6. }
7. function checkGM() {
8.     if (window._GM_xmlHttpRequest) {
9.         window._GM_xmlHttpRequest(
10.            {method: 'GET', url: 'file:///c:/boot.ini'}
11.            onload: function(Response) {
12.                document.formname.textfield.value = Response.responseText;
13.            });
14.     }
15. }
16. if (typeof window.addEventListener != 'undefined') {
17.     window.watch('GM_apis', trapGM);
18.     window.addEventListener('load', checkGM, true);
19. }
20. </script>
```

# JavaScript-level information flow

- Modified Firefox's JavaScript interpreter to track information flow labels
- Sources:
  - Document, Form, History, Passwords, Cookies, Location/Link, Streams, etc.
- Sinks
  - Files/Processes, Network, DOM
- Also had sources/sinks for integrity policies

# Implementation notes

- Several challenges:
  - Precise flow analysis across browser subsystems
  - Tracking instruction provenance
    - Instruction from the browser or from a JSE?
- Overheads:
  - Per-instruction provenance: 6.1x with SunSpider.
  - Without provenance: 42% with SunSpider

# Evaluation highlights

- Tested on 24 JSEs:
  - 4 malicious/vulnerable
  - 20 supposedly benign (from addons.mozilla.org)
- Upon information flow violation:
  - Analyst studies logs and reports violation or produces a declassifier/endorser
  - Flows of sensitive information that pass through declassifier are not reported

# Findings

- Each JSE had features that can be misused
  1. Interaction with HTML forms
  2. Sending/receiving data over an HTTP channel
  3. Interaction with file system
  4. Loading a URL
  5. Communication via JavaScript events
- Tracking instruction provenance is key to avoiding false alerts

# Results with benign JSEs

JSE	Advertised Functionality of JSE	1	2	3	4	5
1. Adblock Plus	Prevent page elements, such as ads, from being downloaded		✓	✓		
2. All-in-One-Sidebar	Sidebar control to switch between sidebar panels and view dialog windows			✓		
3. CoolPreviews	Preview links and images without leaving current page or tab.		✓	✓		
4. Download Statusbar	Manage downloads from a tidy statusbar			✓		
5. Fast Video Download	Easy download of video files from popular sites				✓	
6. Forecastfox	Gets weather forecasts from AccuWeather.com		✓	✓	✓	
7. Foxmarks Synchronizer	Keeps bookmarks and passwords backed up and synchronized		✓	✓		
8. Ghostery	Alerts user's about web bugs, ad networks and widgets on webpages			✓		
9. GooglePreview	Inserts thumbnails and ranks of web sites into Google search results		✓	✓		
10. Greasemonkey (0.8.1)	Allows users customize webpages with user scripts		✓	✓		
11. NoScript	Restricts executable content to trusted domains		✓	✓		
12. PDF Download	Tool for handling, viewing and creating Web-based PDF files		✓	✓	✓	
13. Pwdhash	Customizes user passwords to domains to prevent phishing	✓				
14. SpeedDial	Easy access to frequently visited websites			✓	✓	
15. StumbleUpon	Discovers web sites based on user's interests		✓	✓	✓	
16. Stylish	Easy management of user styles to enhance browsing experience		✓	✓	✓	✓
17. Tab Mix Plus	Enhances Firefox's tab browsing capabilities			✓	✓	
18. User Agent Switcher	Switches the user agent of the browser			✓		
19. Video DownloadHelper	Tool for web content extraction		✓	✓		
20. Web-of-Trust	Warns users before they interact with a harmful site		✓	✓	✓	

Behavior key: (1) HTML forms; (2) HTTP channels; (3) File system; (4) Loading URLs; (5) JavaScript events.

# Take home points

- JSEs pose a threat to browser security
  - Inadequate sandboxing of JSEs
  - Browser and JSE vulnerabilities
- In-browser JavaScript-level information flow tracking can detect security violations
- Supposedly benign JSEs often have suspicious information flows that require whitelisting

# Analyzing Information Flow in JavaScript-based Browser Extensions

Mohan Dhawan and Vinod Ganapathy

Department of Computer Science

Rutgers University

`{mdhawan, vinodg}@cs.rutgers.edu`