Abstract

In this paper, we present a systematic study of how to make a browser secure. Web browser is vulnerable to different attacks; these attacks are performed due to vulnerabilities in the UI of the web page, Browser cache memory, extensions, plug-in. The Attacker can run malicious JavaScript to exploit user system by using these vulnerabilities. Buffer overflow attack, Cross-site-scripting, Man-in-the-middle, Extension vulnerability, Extreme Phishing, Browser Cache poisoning, Session hijacking, Drive-by-download, Click-jacking attacks are discussed. Browser with electrolysis system and sandboxed processes are discussed to prevent the browser from attack.

References

1. Adi, Saltzman, Roi and Sharabani, Active Man in the Middle Attacks: A Security Advisory, A whitepaper from IBM Rational Application Security Group, 2009
2. Bhargava and Chen, Daniel, Shastry, DeFreez, Jean-Pierre Hao and Seifert, A first look at
Firefox OS security, Nashville, TN USA, 2011
3. Xiaowei and Xue, Yuan, Li, A survey on web application security, Nashville, TN USA, 2011

10. David, Stefan, Deian and Yang, Petr and Russo, Edward Z and Marchenko, David and Karp, Alejandro and Herman, Brad and Mazieres, Protecting Users by Confining JavaScript with COWL, (2014)131–146
13. Jerry, Louis, Detection of session hijacking, 2011
17. Adam and Felt, Barth, Adrienne Porter and Saxena, Prateek and Boodman, Aaron, Protecting Browsers from Extension Vulnerabilities, 2010
19. Hodges, Collin and Barth, Jeff and Jackson, Adam, Http strict transport security (hsts), 2012
20. Gu, Boxuan and Zhang, Xiaole and Champion, Wenbin and Bai, Adam C and Qin, Dong, Feng and Xuan, Jsguard: shellcode detection in JavaScript, (2012)112–130
23. Callegati, Walter and Ramilli, Franco and Cerroni, Marco, Man-in-the-Middle Attack to the
24. Eriksson, Mattias and Johansson, TT, An example of a man-in-the-middle attack against server authenticated ssl-sessions, 2003
26. Matthias and Ben-David, Vallentin, Yahel, Persistent browser cache poisoning, 2010
27. Karapanos, Srdjan, Nikolaos and Capkun, On the Effective Prevention of TLS Man-In-The-Middle Attacks in Web Applications, 14, 2014
28. Barth, Adrienne Porter, Adam and Felt, Saxena Prateek and Boodman, Aaron, Protecting Browsers from Extension Vulnerabilities, 2010
30. Vallentin, Yahel, Matthias and Ben-David, Quantifying persistent browser cache poisoning, 2014
35. Deepak Singh, Jain, Divya Rishi and Tomar, Vineeta and Sahu, Session Hijacking: Threat Analysis and Countermeasures
37. Ralf and Basin, Rolf and Hauser, David, Oppliger, SSL/TLS session aware user authentication revisited, Computers Security, 27,(2008)64–70
41. See fixed patches in mozilla on bugzilla, http://www.bugzilla.mozilla.org/quickserack=attachment
44. All errors in Mozilla browser can see one time at, https://www.mozilla.org/en-US/security/known-vulnerabilities/firefox
45. Zhao, Rui and John, Stacy and Bussell, Samantha and Karas, Cara and Roberts, Daniel and Gavett, Jennifer and Six, Brandon and Yue, Chuan, The Highly Insidious Extreme Phishing Attacks,(2016)1–10
46. Privilege escalation vulnerabilities in WebExtensions APIs, https://bugzilla.mozilla.org/showbug.cgi?id=1226423
49. Zhao, Chuan and Yi, Rui and Yue, Qing, Automatic detection of information leakage vulnerabilities in browser extensions, (2015) 1384–1394
50. Integer overflow in Websockets during data buffering,
https://bugzilla.mozilla.org/showbug.cgi?id=1287266
51. Buffer overflow rendering SVG with bidirectional content,
https://bugzilla.mozilla.org/showbug.cgi?id=1270381
52. Cross-site reading attack through data and view-source URIs,
https://bugzilla.mozilla.org/showbug.cgi?id=1228950
53. Integer overflow in MP4 playback in 64-bit versions,
https://bugzilla.mozilla.org/showbug.cgi?id=1206211
54. Same origin violation and local file stealing via PDF reader,
https://bugzilla.mozilla.org/showbug.cgi?id=1178058
57. Electrolysin and multiple content process,
https://wiki.mozilla.org/Electrolysis/Multiplecontentprocesses
58. Sandbox security process model
https://wiki.mozilla.org/Security/Sandbox/Processmodel
59. Hardening the Firefox Security Sandbox
https://wiki.mozilla.org/Security/Sandbox/Hardening
61. Chariton, Argyroudis, Patroklos and Karamitas, Exploiting the jemalloc Memory Allocator: Owning Firefox's Heap, Blackhat USA, 2012

Index Terms
Computer Science                     Security

Keywords
Web application security, Heap overflow, Electrolysis, Sandboxing