the browser. Weinberg et al. created several CAPTCHAs out of the pixels of these links; hence, by answering the CAPTCHA, the user exposed the color of the pixels. In this way, the attacker learned whether the link was followed or not. Our attacks extract much more information from the victim site itself. However, the techniques presented by Weinberg et al. [28] to combine many Boolean questions in one CAPTCHA, can be used to optimize the multiple-questions in malicious CAPTCHA attacks, as described in Section 4.

Unlike the malicious CAPTCHA attack, which is suitable to expose sensitive private information from many websites, Weinberg et al. focused on specific Boolean questions: whether or not the victim browsed to a URL with her browser.

8. CONCLUSIONS

We showed a simple and effective attack that allows attackers to expose private details presented by websites. This is done by exploiting the users themselves as a side-channel, tricking the users into disclosing their own private information. The attack works using all standard browsers, and against some of the most well-known and guarded software-as-a-service web-services, such as Google and Facebook (see more in Table 1).

Similar to other web attacks, defending against this sort of attack is not very difficult, as we explain in Section 6. However, appropriate defenses, especially short term defenses that do not assume new client-side mechanisms, may require web services to slightly modify some mechanisms, e.g., social-network buttons. Such changes may involve a small loss in functionality. It would be interesting to see if and how the industry responds to this challenge: will the small loss in functionality be accepted in order to protect user privacy?

9. ACKNOWLEDGMENTS

This research was supported by a grant from the Ministry of Science and Technology, Israel.

10. REFERENCES

[1] D. Akhawe, W. He, Z. Li, R. Moazzezi, and D. Song. Clickjacking revisited: A perceptual view of UI security. In *8th USENIX Workshop on Offensive Technologies (WOOT 14)*, San Diego, CA, Aug. 2014. USENIX Association.

[2] Alexa. Top Sites. `http://www.alexa.com/topsites`, 2015.

[3] M. Balduzzi, M. Egele, E. Kirda, D. Balzarotti, and C. Kruegel. A solution for the automated detection of clickjacking attacks. In *Proc. of the 5th ASIACCS, ACM Symp. on Information, Computer and Communications Security*, pages 135–144. ACM, 2010.

[4] E. Balsa, C. Troncoso, and C. Diaz. OB-PWS: obfuscation-based private web search. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 491–505. IEEE, 2012.

[5] M. Barbaro and T. Zeller. A face is exposed for AOL searcher no. 4417749. *New York Times*, Aug. 2006.

[6] A. Bortz and D. Boneh. Exposing private information by timing web applications. In *Proceedings of the 16th international conference on World Wide Web*, pages 621–628. ACM, 2007.

[7] Brad Stone. Breaking Google CAPTCHAs for Some Extra Cash. `http://bits.blogs.nytimes.com/2008/03/13/breaking-google-captchas-for-3-a-day/?_r=0`, 2008.

[8] Chester Wisniewski. Facebook adds speed bump to slow down likejackers, March 2011. Online at `https://nakedsecurity.sophos.com/2011/03/30/facebook-adds-speed-bump-to-slow-down-likejackers/`.

[9] M. Egele, L. Bilge, E. Kirda, and C. Kruegel. CAPTCHA smuggling: hijacking web browsing sessions to create CAPTCHA farms. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1865–1870. ACM, 2010.

[10] Eric Lawrence. ClickJacking Defenses. `http://blogs.msdn.com/b/ie/archive/2009/01/27/ie8-security-part-vii-clickjacking-defenses.aspx`, 2009.

[11] Facebook. Social plugins, February 2015. Online at `https://developers.facebook.com/docs/plugins`.

[12] A. J. Ferguson. Fostering e-mail security awareness: The west point carronade. *EDUCASE Quarterly*, 2005.

[13] N. Gelernter and A. Herzberg. Cross-site search attacks. In *Proceedings of the 22nd ACM Conference on Computer and Communications Security*, CCS '15, pages 1394–1405, 2015.

[14] R. Hansen and J. Grossman. Clickjacking. *Sec Theory, Internet Security*, 2008.

[15] A. Herzberg and R. Margulies. Forcing Johnny to login safely. *Journal of Computer Security*, 21(2):393–424, 2013.

[16] B. Hill. Adaptive user interface randomization as an anti-clickjacking strategy, May 2012. `http://www.thesecuritypractice.com/the_security_practice/papers/adaptiveuserinterfacerandomization.pdf`.

[17] L.-S. Huang, A. Moshchuk, H. J. Wang, S. Schecter, and C. Jackson. Clickjacking: Attacks and defenses. In *USENIX Security Symposium*, pages 413–428, 2012.

[18] D. Irani, M. Balduzzi, D. Balzarotti, E. Kirda, and C. Pu. Reverse social engineering attacks in online social networks. In *Detection of intrusions and malware, and vulnerability assessment*, pages 55–74. Springer, 2011.

[19] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer. Social phishing. *Communications of the ACM*, 50(10):94–100, 2007.

[20] Jeremiah Grossman. I Know What Websites You Are Logged-In To (Login-Detection via CSRF). `http://blog.whitehatsec.com/`, 2009.

[21] S. Lekies and M. Heiderich. On the fragility and limitations of current browser-provided clickjacking protection schemes. In E. Bursztein and T. Dullien, editors, *WOOT*, pages 53–63. USENIX Association, 2012.

[22] Mozilla Developer Network. Same-Origin Policy, 2014. `https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy`.

[23] Mozilla Developer Network. Using the application cache, 2015. `https://developer.mozilla.org/en-US/docs/Web/HTML/Using_the_application_cache`.

[24] G. Rydstedt, E. Bursztein, D. Boneh, and C. Jackson. Busting frame busting: a study of clickjacking vulnerabilities at popular sites. In *in IEEE Oakland Web 2.0 Security and Privacy (W2SP 2010)*, pages 1–13, 2010.

[25] The Open Web Application Security Project. Cross-Site Request Forgery. `https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)`, 2010.

[26] T. Van Goethem, W. Joosen, and N. Nikiforakis. The clock is still ticking: Timing attacks in the modern web. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1382–1393. ACM, 2015.

[27] L. Von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum. reCAPTCHA: Human-based Character Recognition via Web Security Measures. *Science*, 321(5895):1465–1468, 2008.

[28] Z. Weinberg, E. Y. Chen, P. R. Jayaraman, and C. Jackson. I still know what you visited last summer: Leaking browsing history via user interaction and side channel attacks. In *Security and Privacy (SP), 2011 IEEE Symposium on*, pages 147–161. IEEE, 2011.