than is typical, we cannot directly infer the maliciousness of a particular app judging from the permissions it requests. Secondly, while our analysis confirms the higher risk with free and mature apps, and that there is a lack of reliable signals, we are not sure if users (in particular mature app users) are actually aware of the privacy risks and making the tradeoff willingly. Studies to examine the privacy tradeoff of users will be interesting. Leveraging on large datasets, we also plan to explore the use of machine learning methods for automatic classification of app privacy intrusiveness.

8. ACKNOWLEDGMENTS

This work has benefited from initial discussions with Adrienne Porter Felt and David Barrera. We are also grateful to Adrienne and the anonymous reviewers for their valuable comments on earlier drafts.

9. REFERENCES

[1] A. Acquisti and R. Gross. Imagined communities: Awareness, information sharing, and privacy on the facebook. In G. Danezis and P. Golle, editors, *Privacy Enhancing Technologies*, volume 4258 of *Lecture Notes in Computer Science*, pages 36–58. Springer, 2006.

[2] Android Developer's Guide – Manifest Permissions. `http://developer.android.com/reference/android/Manifest.permission.html`.

[3] Android Market. `https://market.android.com`.

[4] AppBrain. `http://www.appbrain.com`.

[5] D. Barrera, W. Enck, and P. C. van Oorschot. Seeding a Security-Enhancing Infrastructure for Multi-market Application Ecosystems. Technical report, Carleton University, April 2011. TR-11-06.

[6] D. Barrera, P. C. van Oorschot, and A. Somayaji. A Methodology for Empirical Analysis of Permission-Based Security Models and its Application to Android Categories and Subject Descriptors. In *Proc. of the 17th ACM conf. on Computer and Communications Security*, CCS '10, pages 73–84. ACM, 2010.

[7] J. Bonneau, J. Anderson, and L. Church. Privacy suites: shared privacy for social networks. In *Proc. of the 5th Symposium on Usable Privacy and Security*, SOUPS '09. ACM, 2009.

[8] P. H. Chia, A. P. Heiner, and N. Asokan. Use of ratings from personalized communities for trustworthy application installation. In *Proc. of the 15th Nordic conf. in Secure IT Systems*, NordSec '10, 2010.

[9] P. H. Chia and S. J. Knapskog. Re-evaluating the wisdom of crowds in assessing web security. In G. Danezis, editor, *Financial Cryptography and Data Security, FC '11*, volume 7035 of *Lecture Notes in Computer Science*, pages 299–314. Springer, 2012.

[10] F. Cohen. Computational aspects of computer viruses. *Computers & Security*, 8(4):297–298, 1989.

[11] F. J. Damerau. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7:171–176, March 1964.

[12] W. Enck, M. Ongtang, and P. McDaniel. On lightweight mobile phone application certification. In *Proc. of the 16th ACM conf. on Computer and Communications Security*, CCS '09, pages 235–245. ACM, 2009.

[13] Facebook Developers – Permissions. `https://developers.facebook.com/docs/reference/api/permissions/`.

[14] Facebook partners with WOT. Article on ArcticStartup website, May 2011. `http://www.arcticstartup.com/2011/05/12/facebook-partners-with-wot-to-protect-its-700-million-users`.

[15] A. P. Felt. Personal Communication.

[16] A. P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner. Android permissions demystified. In *Proc. of the 18th ACM conf. on Computer and Communications Security*, CCS '11, pages 627–638. ACM, 2011.

[17] A. P. Felt, K. Greenwood, and D. Wagner. The effectiveness of application permissions. In *Proc. of the 2nd USENIX conf. on Web application development*, WebApps '11. USENIX Association, 2011.

[18] Google Chrome Extensions – Permission Warnings. `http://code.google.com/chrome/extensions/permission_warnings.html`.

[19] Google Chrome Web Store – Extensions. `https://chrome.google.com/webstore?category=ext`.

[20] J. King, A. Lampinen, and A. Smolen. Privacy: Is there an app for that? In *Proc. of the 7th Symposium on Usable Privacy and Security*, SOUPS '11, pages 12:1–12:20. ACM, 2011.

[21] K. Kostiainen, E. Reshetova, J.-E. Ekberg, and N. Asokan. Old, new, borrowed, blue –: a perspective on the evolution of mobile platform security architectures. In *Proc. of the 1st ACM conf. on Data and Application Security and Privacy*, CODASPY '11, pages 13–24. ACM, 2011.

[22] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.

[23] M. Marsall. How HTML5 will kill the native app. Article on VentureBeat website, April 2011. `http://venturebeat.com/2011/04/07/how-html5-will-kill-the-native-app/`.

[24] T. Moore and B. Edelman. Measuring the perpetrators and funders of typosquatting. In R. Sion, editor, *Financial Cryptography and Data Security, FC '10*, volume 6052 of *Lecture Notes in Computer Science*, pages 175–191. Springer, 2010.

[25] Our project site. `http://aurora.q2s.ntnu.no/app`.

[26] Socialbakers – Applications on Facebook. `http://www.socialbakers.com/facebook-applications`.

[27] J. Tam, R. W. Reeder, and S. Schechter. I'm Allowing What? Disclosing the authority applications demand of users as a condition of installation. Technical report, Microsoft Research, 2010. MSR-TR-2010-54.

[28] Watir – Web Application Testing in Ruby. `http://watir.com`.

[29] Web of Trust (WOT). `http://www.mywot.com`.

[30] WhatApp? A Stanford Center for Internet and Society website. `https://whatapp.org/`.

[31] D. M. Wilkinson. Strong regularities in online peer production. In *Proc. of the 9th ACM conf. on Electronic commerce*, EC '08, pages 302–309. ACM, 2008.