would, however, be one of the key benefits of having a secure element inside a mobile phone.

V. Reporting and Industry Response

We reported our findings and proposed workarounds to Google (and some of their Google Wallet partners) in April 2012. Google quickly acknowledged the problem and confirmed that they could reproduce the attack. Our tests in June 2012 revealed that new installations of Google Wallet (i.e. secure element applets provisioned in June) were no longer vulnerable to our relay attack setup. Further testing in September 2012 showed that users of older versions of Google Wallet are now required to update to the latest version. This forces existing users to receive the necessary fixes of the secure element applets. Therefore, we assume that Google Wallet users are no longer vulnerable to the relay attack scenario described in this paper.

VI. Analysis of the Relay-immune Google Wallet

Version 1.6 of the Google Wallet on-card component (installed with version 1.5-R79-v5 of Google Wallet in September 2012) is no longer vulnerable to the software-based relay attack setup described in this paper. The relay attack is inhibited by the fact that the select command fails for both MasterCard credit card applet instances (`A000000004 1010` and `A00000 0004 1010 AA539648FFFF00FFFF`) with the error code `6999`. Thus, access to the credit card applet from the application processor has been disabled as we suggested (cf. section IV-D3). The Proximity Payment System Environment can still be selected though both internal and external mode.

The on-card component can still only be selected through internal mode. It now returns its version number upon selection and some commands for interaction with it have slightly changed their parameters. The commands for switching between locked and unlocked state of the wallet are still the same. As a result, it is still possible to unlock Google Wallet and the credit card contained in it without PIN verification. Consequently, a malicious application could enable the credit card on the RF contactless interface even though Google Wallet is protected by a PIN that is not known to the malicious application.

VII. Conclusion

In this paper we examined the feasibility of the software-based relay attack based on the mobile contactless payment application Google Wallet. We analyzed the communication between the Google Wallet app and the secure element, as well as the interaction between a point-of-sale credit card terminal and the Google Wallet device. Then, we used this information to create a prototype relay setup. With this setup we could confirm that Google Wallet was indeed vulnerable to the software-based relay attack. On the one hand, the credit card applets in the secure element were not sufficiently protected from access through apps on the application processor. On the other hand, the PIN protection of Google Wallet can be bypassed as the on-card component does not verify the PIN itself but instead can be controlled by simple lock and unlock commands. Google responded to our finding by fixing the vulnerability to software-based relay attacks. However, bypassing the PIN is still possible with the current version of the wallet.

Acknowledgment

This work is part of the project “4EMOBILITY” within the EU programme “Regionale Wettbewerbsfähigkeit OÖ 2007–2013 (Regio 13)” funded by the European regional development fund (ERDF) and the Province of Upper Austria (Land Oberösterreich).

References

[1] M. Roland, “Applying recent secure element relay attack scenarios to the real world: Google Wallet Relay Attack,” Technical Report, arXiv:1209.0875 [cs.CR], Sep. 2012, http://arxiv.org/abs/1209.0875.

[2] A. Hoog, “Forensic security analysis of Google Wallet,” *viaForensics Mobile Security Blog*, Dec. 2011, https://viaforensics.com/mobile-security/forensics-security-analysis-google-wallet.html.

[3] C. Benninger, “A Brave New Wallet – First look at decompiling Google Wallet,” *Intrepidus Group Insight*, Sep. 2011, http://intrepidusgroup.com/insight/2011/09/a-brave-new-wallet-first-look-at-decompiling-google-wallet/.

[4] N. Elenkov, “Exploring Google Wallet using the secure element interface,” *Android Explorations*, Aug. 2012, http://nelenkov.blogspot.com/2012/08/exploring-google-wallet-using-secure.html.

[5] N. Elenkov, “Accessing the embedded secure element in Android 4.x,” *Android Explorations*, Aug. 2012, http://nelenkov.blogspot.com/2012/08/accessing-embedded-secure-element-in.html.

[6] N. Elenkov, “Android secure element execution environment,” *Android Explorations*, Aug. 2012, http://nelenkov.blogspot.com/2012/08/android-secure-element-execution.html.

[7] M. Roland, J. Langer, and J. Scharinger, “Practical Attack Scenarios on Secure Element-enabled Mobile Devices,” in *Proceedings of the Fourth International Workshop on Near Field Communication (NFC 2012)*, Helsinki, Finland, Mar. 2012, pp. 19–24.

[8] J. Rubin, “Google Wallet Security: PIN Exposure Vulnerability,” *zveloBLOG*, Feb. 2012, https://zvelo.com/blog/entry/google-wallet-security-pin-exposure-vulnerability.

[9] G. P. Hancke, “A Practical Relay Attack on ISO 14443 Proximity Cards,” Jan. 2005, http://www.rfidblog.org.uk/hancke-rfidrelay.pdf.

[10] Z. Kfir and A. Wool, “Picking Virtual Pockets using Relay Attacks on Contactless Smartcard,” in *Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM’05)*, Sep. 2005, pp. 47–58.

[11] L. Francis, G. P. Hancke, K. E. Mayes, and K. Markantonakis, “Practical NFC Peer-to-Peer Relay Attack Using Mobile Phones,” in *Radio Frequency Identification: Security and Privacy Issues*, ser. LNCS. Springer Berlin Heidelberg, 2010, vol. 6370/2010, pp. 35–49.

[12] L. Francis, G. P. Hancke, K. E. Mayes, and K. Markantonakis, “Practical Relay Attack on Contactless Transactions by Using NFC Mobile Phones,” Cryptology ePrint Archive, Report 2011/618, 2011, http://eprint.iacr.org/2011/618.

[13] M. Roland, “Software Card Emulation in NFC-enabled Mobile Phones: Great Advantage or Security Nightmare?” in *4th International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use*, Newcastle, UK, Jun. 2012, http://www.medien.ifi.lmu.de/iwssi2012/papers/iwssi-spmu2012-roland.pdf.

[14] G. P. Hancke, K. E. Mayes, and K. Markantonakis, “Confidence in smart token proximity: Relay attacks revisited,” *Computers & Security*, vol. 28, no. 7, pp. 615–627, 2009.

[15] M. Roland, J. Langer, and J. Scharinger, “Relay Attacks on Secure Element-enabled Mobile Devices: Virtual Pickpocketing Revisited,” in *Information Security and Privacy Research*, ser. IFIP AICT. Springer Boston, Jun. 2012, vol. 376/2012, pp. 1–12.

[16] J. Rubin, “Google Wallet Security: About That Rooted Device Requirement...” *zveloBLOG*, Feb. 2012, https://zvelo.com/blog/entry/google-wallet-security-about-that-rooted-device-requirement.

[17] *EMV Contactless Specifications for Payment Systems – Book B: Entry Point Specification*, EMVCo Spec., Version 2.1, Mar. 2011.

[18] S. Höbarth and R. Mayrhofer, “A framework for on-device privilege escalation exploit execution on Android,” in *3rd International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use*, San Francisco, CA, USA, Jun. 2011.

[19] “Google Wallet – How it works – In-store,” http://www.google.com/wallet/how-it-works/in-store.html, Sep. 2012.