Springer

Low-Level Attacks in Bitcoin Wallets

International Conference on Information Security
ISC 2017: Information Security pp 233-253 | Cite as

Conference paper

First Online: 20 October 2017

Part of the Lecture Notes in Computer Science book series (LNCS, volume 10599)

Abstract

As with every financially oriented protocol, there has been a great interest in studying, verifying, attacking, identifying problems, and proposing solutions for Bitcoin. Within that scope, it is highly recommended that the keys of user accounts are stored offline. To that end, companies provide solutions that range from paper wallets to tamper-resistant smart-cards, offering different level of security. While incorporating expensive hardware for the wallet purposes is though to bring guarantees, it is often that the low-level implementations introduce exploitable back-doors. This paper aims to bring to attention how the overlooked low-level protocols that implement the hardware wallets can be exploited to mount Bitcoin attacks. To demonstrate that, we analyse the general protocol behind Ledger Wallets, the only EAL5+ certified against side channel analysis attacks hardware. In this work we conduct a throughout analysis on the Ledger Wallet communication protocol and show how to successfully attack it in practice. We address the lack of well-defined security properties that Bitcoin wallets should conform by articulating a minimal threat model against which any hardware wallet should defend. We further use that threat model to propose a lightweight fix that can be adopted by different technologies.

A Appendix

A.1 Example Communication Trace

As an example, we provide the trace that was generated for the following transaction:

Transaction Id	92d30a91b45d6ab528af12f3a9c0701e01f67348a257ed50362439a2ee8274e7
Input addresses	**1** 113biVTVQk73Eem1UYYn9YcrPVrxp6xeVc **2** 15DpocdQpwXeUp9Ccf2Nz9AQ9jKp9U5VdZ
Payment address	1GocNQ4Q8BtzacpHQiGLWk9vNppoq6Lh8W
Payment amount	0.00813844
Change address	1PmXm9UcAgDBp5i3SvqD3SfdKChfWthH4W

We only provide the traces of the commands of Fig. 4 so as not to overwhelm.

1. `get_trusted_input`: `e042000009000000010100000001`

2. `response`: `9000`

3. `get_trusted_input`: `e04280000400000000`

4. `response`: `32008ed5f038879105a5778cdacee02ca43f21bcbbd66cd647add3db69dd3222b9c3968`
`d0000000078710d00000000009132801b579e659b`

5. `get_wallet_public_key`: `e040000015058000002c80000000800000000000000000000000c`

6. `response`: `410441ec4b255d40010284f117d8105456a268cd9536ca5ca3d3016bf6d21902e5dc4bf`
`9b224b5cb2379b5c2b4a47044862d42c6e5b14daf22939fec8023c83ac519223131336269565456 51`
`6b373345656d315559596e39596372505672787036786556 63da55cec9398694400832d6af2426c05`
`7addc73438efa016f6f9232735ee6b1a8`

7. `get_wallet_public_key`: `e040000015058000002c800000008000000000000000100000012`

8. `response`: `41043f07a649a72651f10d5728b7f848ee879fb3b263ddd653b51b563a051f138fa3e35`
`f5f6d794a2621fbf0493d6af5c2b300734086fa0ebbe411f11017b1989bdd22313544706f63645170`
`77586555703943636632 4e7a394151396a4b7039553556645a9bf32153ef7f646d1d1991382932bc9`
`15d671ddc3640ef8da3eb54877191e559`

9. `untrusted_hash_transaction_input_start`: `e0440000050100000002`

10. `response`: `9000`

11. `untrusted_hash_transaction_input_finalize`: `e046020048223147 6f634e5134513842747a61`
`6370485169474c576b39764e70706f71364c68385700000000000000d6d80000000000000450805800 00`
`02c800000008000000000000000100000013`

12. `response`: `4502d8d60000000000001976a914ad5a8ba5325b4b836c49b09797cbb83744a7a2f588a`
`c146b0c0000000000001976a914f9bebf6735e688877e409cd494ad820b344dd76e88ac03040405121e`
`47646f813e5dfd4fbc72e6698cc40a67a980bccbe7881c2e40ac6fec4fbcda20d980ec3a67445e48d`
`ad870ee58d006745fdf953138be5fb0570e679f512c36ed`

13. `untrusted_hash_sign`: `e04800001f058000002c80000000800000000000000000000000c04040606`
`020000000001`

14.

```
response: 3044022033128d0d576487e2e0c5892c0915564a6a5f119e698c033262d660527943a16
d022009caa037703d9a3dbf7eec4cecca08bf33b3b9a18ef929a810f8faf6ab0f1c7a01
```

A.2 Active Attacks

a.4: Alter the Wallet Security Properties. The attack requires sending the wrong pin p' three consecutive times and then tampering the `set_operation` command. A sample trace with the breakdown of the steps and their corresponding commands is given in Table 6; we underline the important pieces of the exchange.

Table 6.

Attack a.4: Trace of disabling the second factor authentication during *Setup*

Steps	APDU traces
Block the dongle	• verify(p'): 0220000004<u>33333333</u> • verify(p'): 0220000004<u>33333333</u> • verify(p'): 0220000004<u>33333333</u>
Replay a Setup Session	• setup(p, s): e02000004c020a0005<u>0431343234</u> 00408c3937fafb22e5f4979e90afe0b912cc05d92b9910c622887f61b30d9814f714df2dd5ada8cc5cd663e998dec1cc55915377352cf6949a20ba444039219efd6900 • set_keyboard: e028000077000000000000000000000000760f00d4ffffffc7000000782c1e3420212224342627252e362d3738271e1f202122232425263333362e37381f0405060708090a0b0c0d0e0f101112131415161718191a1b1c1d2f3130232d350405060708090a0b0c0d0e0f10111213141516171819 1a1b1c1d2f313035 • get_device_attestation: e0c200000861255ccee7f8c72d • set_operation: <u>e02600000102</u> ...

b.1–b.2: Transaction Attacks. The structure of `untrusted_hash_transaction_input_finalize` is:

command	e046020048
length of payment address	22
payment address $addr_p$	314e33717572335965653334b664e74436a4677756e346f366f4c324478686747796
payment amount $amount_p$	0000000000005305
fees $fees_p$	0000000000001d60
change address BIP32 parameters	058000002c80000000800000000000000001000000
second authentication status (true/false)	02

and the structure of the response data that we are interested in is:

payment amount $amount_p$	`03b1000000000000`
hash160 of $addr_p$	`f1253f0463e5877c5e8bb3f34e7abfb335023ee1`
change c	`0553000000000000`
hash160 change address $addr_c$	`e6e44d66125327341d6abb71e0702a4ea0537437`

Depending on the attack we want to perform the corresponding data part needs to be altered. For example, to change the payment address from `163WPEeTHjvFsUfx1UbDPXK92eRmqXQrGA` to `113biVTVQk73Eem1UYYn9YcrPVrxp6xeVc`, we tamper the original command:

`e04602004822` `3136335750456554486a7646735566783155624450584b3932655` `26d715851724741` `00000000000027100000000000001a9a058000002c80000000800000000000000100000000` to the command:

`e04602004822` `3131336269565456516b373345656d315559596e395963725056` `7278703678655663` `00000000000027100000000000001a9a058000002c80000000800000000000000100000000`

where we underline the relevant parts; similarly for the response.

Learning the Security Card. The attacker gains access to the keycard mappings, *secFR*, via the `untrusted_hash_sign` command, *e.g.*, `e04800001f058000002c80000000800000000000000000000000104` `0f090a02` `0000000001`.

References

1. Androulaki, E., Karame, G.O., Roeschlin, M., Scherer, T., Capkun, S.: Evaluating user privacy in bitcoin. In: Sadeghi, A.-R. (ed.) FC 2013. LNCS, vol. 7859, pp. 34–51. Springer, Heidelberg (2013). doi: 10.1007/978-3-642-39884-1_4 (https://doi.org/10.1007/978-3-642-39884-1_4)
   CrossRef
   Google Scholar
2. Bamert, T., Decker, C., Wattenhofer, R., Welten, S.: BlueWallet: the secure bitcoin wallet. In: Mauw, S., Jensen, C.D. (eds.) STM 2014. LNCS, vol. 8743, pp. 65–80. Springer, Cham (2014). doi: 10.1007/978-3-319-11851-2_5 (https://doi.org/10.1007/978-3-319-11851-2_5)
   Google Scholar
3. Barber, S., Boyen, X., Shi, E., Uzun, E.: Bitter to better — how to make bitcoin a better currency. In: Keromytis, A.D. (ed.) FC 2012. LNCS, vol. 7397, pp. 399–414. Springer, Heidelberg (2012). doi: 10.1007/978-3-642-32946-3_29 (https://doi.org/10.1007/978-3-642-32946-3_29)
   CrossRef
   Google Scholar
4. Biham, E., Shamir, A.: Differential fault analysis of secret key cryptosystems. In: Kaliski, B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 513–525. Springer, Heidelberg (1997). doi: 10.1007/BFb0052259 (https://doi.org/10.1007/BFb0052259)
   CrossRef
   Google Scholar
5. Bitcoin ewallet vanishes from internet. http://www.tribbleagency.com/?p=8133 (http://www.tribbleagency.com/?p=8133)
6. Bitcoin Protocol Documentation. https://en.bitcoin.it/wiki/Protocol%5Fdocumentation (https://en.bitcoin.it/wiki/Protocol%5Fdocumentation)
7. Bitcoinmagazine (2013). https://bitcoinmagazine.com/articles/ozcoin-hacked-stolen-funds-seized-and-returned-by-strongcoin-1366822516 (https://bitcoinmagazine.com/articles/ozcoin-hacked-stolen-funds-seized-and-returned-by-strongcoin-1366822516)
8. Bozzato, C., Focardi, R., Palmarini, F., Steel, G.: `APDU`-level attacks in `PKCS#11` devices. In: Monrose, F., Dacier, M., Blanc, G., Garcia-Alfaro, J. (eds.) RAID 2016. LNCS, vol. 9854, pp. 97–117. Springer, Cham (2016).

doi: 10.1007/978-3-319-45719-2_5 (https://doi.org/10.1007/978-3-319-45719-2_5)
CrossRef
Google Scholar

9. Datko, J., Quartier, C., Belyayev, K.: Breaking bitcoin hardware wallets. In: DEFCON (2017)
Google Scholar
10. De Koning Gans, G., De Ruiter, J.: The smartlogic tool: analysing and testing smart card protocols. In: 2012 IEEE Fifth International Conference on Software Testing, Verification and Validation (ICST), pp. 864–871 (2012)
Google Scholar
11. Decker, C., Wattenhofer, R.: Bitcoin transaction malleability and MtGox. In: Kutyłowski, M., Vaidya, J. (eds.) ESORICS 2014. LNCS, vol. 8713, pp. 313–326. Springer, Cham (2014). doi: 10.1007/978-3-319-11212-1_18 (https://doi.org/10.1007/978-3-319-11212-1_18)
Google Scholar
12. Genkin, D., Shamir, A., Tromer, E.: RSA key extraction via low-bandwidth acoustic cryptanalysis. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014. LNCS, vol. 8616, pp. 444–461. Springer, Heidelberg (2014). doi: 10.1007/978-3-662-44371-2_25 (https://doi.org/10.1007/978-3-662-44371-2_25)
CrossRef
Google Scholar
13. Gkaniatsou, A., McNeill, F., Bundy, A., Steel, G., Focardi, R., Bozzato, C.: Getting to know your card: reverse-engineering the smart-card application protocol data unit. In: Proceedings of the 31st Annual Computer Security Applications Conference, pp. 441–450 (2015)
Google Scholar
14. Hao, F., Ryan, P.: J-PAKE: authenticated key exchange without PKI. In: Gavrilova, M.L., Tan, C.J.K., Moreno, E.D. (eds.) Transactions on Computational Science XI. LNCS, vol. 6480, pp. 192–206. Springer, Heidelberg (2010). doi: 10.1007/978-3-642-17697-5_10 (https://doi.org/10.1007/978-3-642-17697-5_10)
CrossRef
Google Scholar
15. Herrera-Joancomartí, J.: Research and challenges on bitcoin anonymity. In: Garcia-Alfaro, J., Herrera-Joancomartí, J., Lupu, E., Posegga, J., Aldini, A., Martinelli, F., Suri, N. (eds.) DPM/QASA/SETOP -2014. LNCS, vol. 8872, pp. 3–16. Springer, Cham (2015). doi: 10.1007/978-3-319-17016-9_1 (https://doi.org/10.1007/978-3-319-17016-9_1)
Google Scholar
16. Higgins, S. (2015). http://www.coindesk.com/unconfirmed-report-5-million-bitstamp-bitcoin-exchange (http://www.coindesk.com/unconfirmed-report-5-million-bitstamp-bitcoin-exchange)
17. Hsiao, H.-C., Lin, Y.-H., Studer, A., Studer, C., Wang, K.-H., Kikuchi, H., Perrig, A., Sun, H.-M., Yang, B.-Y.: A study of user-friendly hash comparison schemes. In: Annual Computer Security Applications Conference, ACSAC 2009, pp. 105–114. IEEE (2009)
Google Scholar
18. Huang, D.Y., Dharmdasani, H., Meiklejohn, S., Dave, V., Grier, C., McCoy, D., Savage, S., Weaver, N., Snoeren, A.C., Levchenko, K.: Botcoin: monetizing stolen cycles. In: 21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, California, USA, 23–26 February 2014
Google Scholar
19. Karame, G.O., Androulaki, E., Capkun, S.: Double-spending fast payments in bitcoin. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security, CCS 2012, pp. 906–917 (2012)
Google Scholar
20. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999). doi: 10.1007/3-540-48405-1_25 (https://doi.org/10.1007/3-540-48405-1_25)
Google Scholar
21. Kocher, P.C.: Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 104–113. Springer, Heidelberg (1996). doi: 10.1007/3-540-68697-5_9 (https://doi.org/10.1007/3-540-68697-5_9)
Google Scholar
22. Lim, I.-K., Kim, Y.-H., Lee, J.-G., Lee, J.-P., Nam-Gung, H., Lee, J.-K.: The analysis and countermeasures on security breach of bitcoin. In: Murgante, B., Misra, S., Rocha, A.M.A.C., Torre, C., Rocha, J.G., Falcão, M.I., Taniar, D., Apduhan, B.O., Gervasi, O. (eds.) ICCSA 2014. LNCS, vol. 8582, pp. 720–732. Springer, Cham (2014). doi: 10.1007/978-3-319-09147-1_52 (https://doi.org/10.1007/978-3-319-09147-1_52)
Google Scholar
23. Murdoch, S.J., Drimer, S., Anderson, R.J., Bond, M.: Chip and PIN is broken. In: 31st IEEE Symposium on Security and Privacy, S&P 2010, Berleley/Oakland, California, USA, 16–19 May 2010, pp. 433–446 (2010)
Google Scholar
24. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system. http://bitcoin.org/bitcoin.pdf (http://bitcoin.org/bitcoin.pdf)
25. Poulsen, K.: New malware steals your bitcoin (2011). https://www.wired.com/2011/06/bitcoin-malware (https://www.wired.com/2011/06/bitcoin-malware)
26. Rosenfeld, M.: Analysis of hashrate-based double spending. CoRR, abs/1402.2009 (2014)
Google Scholar
27. The Bitcoin Wiki. https://en.bitcoin.it/wiki/Wallet%5Fencryption (https://en.bitcoin.it/wiki/Wallet%5Fencryption)
28. The Bitcoin Wiki (2014). https://en.bitcoin.it/wiki (https://en.bitcoin.it/wiki)
29. Turuani, M., Voegtlin, T., Rusinowitch, M.: Automated verification of electrum wallet. In: Clark, J., Meiklejohn, S., Ryan, P.Y.A., Wallach, D., Brenner, M., Rohloff, K. (eds.) FC 2016. LNCS, vol. 9604, pp. 27–42. Springer, Heidelberg

(2016). doi: 10.1007/978-3-662-53357-4_3 (https://doi.org/10.1007/978-3-662-53357-4_3)
CrossRef
Google Scholar

30. Uzun, E., Karvonen, K., Asokan, N.: Usability analysis of secure pairing methods. In: Dietrich, S., Dhamija, R. (eds.) FC 2007. LNCS, vol. 4886, pp. 307–324. Springer, Heidelberg (2007). doi: 10.1007/978-3-540-77366-5_29 (https://doi.org/10.1007/978-3-540-77366-5_29)
CrossRef
Google Scholar
31. Wuille, P.: Dealing with maellability. Online specification for BIP62 (2014)
Google Scholar
32. Wuille, P.: Hierarchical deterministic wallets. Online specification for BIP32 (2017)
Google Scholar

Copyright information

About this paper

Cite this paper as:
Gkaniatsou A., Arapinis M., Kiayias A. (2017) Low-Level Attacks in Bitcoin Wallets. In: Nguyen P., Zhou J. (eds) Information Security. ISC 2017. Lecture Notes in Computer Science, vol 10599. Springer, Cham

DOI (Digital Object Identifier)
https://doi.org/10.1007/978-3-319-69659-1_13

Publisher Name
Springer, Cham

Print ISBN
978-3-319-69658-4

Online ISBN
978-3-319-69659-1

eBook Packages
Computer Science

About this book

Reprints and Permissions