# Investigating the Attack Surface of Hardware Wallets

1st Talha Zia Muhammad Ali
*Embedded Systems and Internet of Things*
*Technical University of Munich*
Munich, Germany
ge46tun@mytum.de

***Abstract*—The safety of cryptographic keys determines the security of cryptocurrencies. Using the innovative concept of blockchains, there is no straightforward way to tamper with online shared transaction ledger. Instead, attackers are interested in cryptographic keys which are necessary for managing cryptographic operations. Hardware wallets are considered the safest way to manage cryptographic key pairs.**

**In this paper we are analyzing the vulnerabilities of hardware wallets in domain of supply chain, hardware, and software architecture. The analysis also inculcates both open-source code and closed source code hardware wallets. In addition, we are also assessing the attacks in terms of severity and scalability. In the end we are concluding that security of hardware wallet remains questionable in specific case of hardware vulnerabilities (side channel attacks), giving a broader picture of future challenges, making hardware wallets resistant to side channel attacks and introduction of social recovery features such that assets can be recovered in case of security breach.**

***Index Terms*—Cryptocurrency, Blockchain, Hardware Wallet, Side Channel Attack, Deep Learning, Private Key.**

## I. Introduction

Innovation of the Nakamoto consensus protocol gave compute nodes the opportunity to agree upon a shared ledger state in a trust-less network setup [1]. Almost all cryptocurrency, including Bitcoin, Ethereum, Bitcoin cash and Litecoin, make use of transparent ledger that is secured via block chain network. Without a central authority, everyone is allowed to connect a compute node to the network. Blockchains allow web services around blockchain nodes, making it feasible for users to interact with blockchains individually. When users send transactions to the blockchain, blockchain nodes order, include, and eventually append transactions to the blockchain state via consecutive blocks. Every transaction change the blockchain state at specific addresses [2]. Ownership of addresses is thereby managed by keys, which are in turn managed by user-controlled wallets.

With increasing value of assets stored at specific blockchain addresses, wallets controlling these addresses became targets for attackers [3]. Advent of software wallet attacks led to the development of hardware wallets which protect cryptographic keys in separate dedicated hardware devices. Even though hardware wallets protect users against many issues [4], hardware wallets are becoming a target itself. Getting a clear overview over recent hardware wallet attacks, their severity, complexity, and structure has found little attention in academic research. Hence, it is difficult to grasp state-of-the-art open challenges around hardware wallet security.

This work tries to clarify the vulnerabilities of hardware wallet keeping in view its hardware and software architecture. This work identifies two major open challenges, making hardware wallets resistant to side channel attacks and introduction of social recovery features such that assets can be recovered in case of security breach. These allow us to conclude that research in field of side channel data prediction of hardware wallet is paramount for security and reliability. This paper is constituted of six sections structured as follows: Section II provides the description of cryptocurrency fundamentals. Section III compares the scope of this research with previous similar work. Section IV covers security analysis of hardware wallets keeping in view recent case studies. The evaluation is covered in Section V, and section VI concludes with a conclusion and future challenges.

## II. Background Study

Cryptocurrency operations is managed and validated with asymmetric cryptography. The private key is used for generation of the corresponding public key and digital signatures on transaction requests as a proof of ownership and validation. The public key is used as an address besides providing its cryptographic purpose of operating in cryptographic algorithms [5]. Considering algorithm of asymmetric cryptography, the private key cannot be reconstructed from a public key or address.

### A. Blockchain Wallets

The wallet is a software that holds a private key and automates complex cryptography. The wallet accepts the requested transaction, signs it on your behalf using your private key, and sends it to a single blockchain node [2]. After validation with digital signature, transaction is entered by the miners on the transaction ledger.

### B. Blockchain Wallets Classification

Blockchain wallet or cryptocurrency wallets can be classified based on:

- Architecture (Standard or Hierarchical Deterministic)
- Functionality (Full Node or Simple Payment Verification)
- Accessibility (Hot Wallet or Cold Wallet)

The following sub-paragraphs refer to the most common types of wallets:

**Standard Wallet:** creates a wallet.dat file that contains a private key [6]. This file must be backed up by copying it to a safe digital storage.

**Hierarchical Deterministic (HD) Wallet:** generates an initial seed phrase, a string of common words that can be memorized instead of long confusing private key. [7].

**Full Node Wallet:** holds full copy of blockchain for validating each and every transaction [8].

**Simple Payment verification (SPV) Wallet:** relies on blockchain nodes that are connected to validate transactions making them faster and lesser memory allocation [8].

**Hot wallet:** has internet access either via web service or a wallet installed on a device connected to internet such that there is a data traffic between host and web network [9].

**Cold wallet:** has no internet access in any way. Cold wallet is also referred to as Hardware wallet or Offline wallet [9].

### *C. Advantages and Disadvantages of Software Wallets*

The web wallet or software wallet is a least secure option for storage since you are asking someone to hold your keys for you. Web wallets are highly convenient by providing high flexibility, allowing you to buy, sell or transfer cryptocurrency at any point and time [10].

### *D. Hardware Wallets*

Consider a computer, running a blockchain wallet, can get infected with malware, exposing private key to bad actors. This can be avoided by:

- Computer is malware free (not practically possible)
- Dedicated single purpose simplified computer to run blockchain wallet

Hardware wallet is a computer that is stripped down of all logic except for a small screen, few buttons, and bare necessities for simple action of storing private key and signing transactions. Lesser complexity makes hardware wallet so dumb that it is theoretically impossible to hack or infect it with anything. Since Hardware wallet is such a simple device that can only store private key and sign transactions, it needs to be connected to a more sophisticated computer. A bridge software is required that can prepare transaction, communicate with hardware wallet, receive digitally signed transaction from hardware wallet and interact with block chain node via internet connectivity. Once a transaction request is received via the bridge software, hardware wallet signs it and sends it back to bridge software. Private key never leaves the hardware wallet. Minimalist and plain design enable them to be used with any computer without fear of being hacked or infected [11].

## III. Related Work

Majority of previous research has focused on either the hardware wallet's specific attack surface (e.g., side channel attacks [12], software attacks [13]) or device-related attacks (TREZOR [3] or Keepkey [14]). However, a work presented at the USENIX Security Symposium in August 2021 [15] provided an interesting perspective by listing potential attack surfaces for hardware wallets and evaluating them on all reputable hardware wallet devices. In addition, a study published in November 2021 [16] focuses on the characteristics and security engineering techniques (firmware attestation, software and hardware attestation). This seminar paper investigates the vulnerabilities of hardware wallets in the supply chain, hardware, and software architectural domains considering recent attacks. This work provides a fair summary as well as an interesting introduction paper to read. This study considers not just scholarly literature but also manufacturer's blog posts, giving readers a unique viewpoint on weaknesses as well as motivation for future research and development.

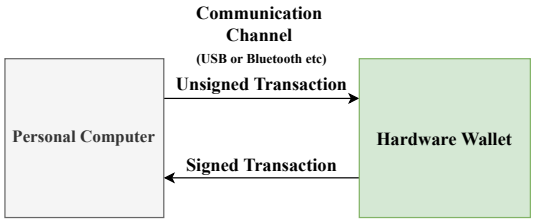


Fig. 1: Hardware Wallet Connectivity with PC

## IV. Security Analysis of Hardware Wallets

Seed phrase is generated by device at random on initialization. Subsection A is a pre-analysis of the hardware wallets keeping in-view the hardware and software architecture. In subsections B, C and D, we go into deeper details of attacks. Attack surfaces are analyzed based on (i) supply chain attacks, (ii) hardware attacks, and (iii) software attacks. Under hardware attacks, this work investigates side channel and deep learning-based attacks. Software attacks are categorized on their scope towards application logic, general data attacks and communication attacks.

### *A. Hardware and Software Architecture*

The hardware wallet can have multiple microcontrollers (MCU) and external peripherals. Hardware wallet software code can be fully open source or closed source. Primarily, hardware wallet provides (i) security enclave such that sensitive elements never leave the device, (ii) compact application programming Interface (API) comparable with microcontroller (MCU) persistent flash storage, and (iii) memory protection unit (MPU).

Single Chip and Dual Chip Design: Hardware wallets may use single general purpose microcontroller as in TREZOR [3] or a dual microcontroller as in Ledger Nano S [11]. Single chip design is just a simple microcontroller, which is not meant to be very secure but fulfils the purpose (Fig 2). Dual chip design consists of a standard microcontroller, used as a proxy for managing the screen, button and connectivity and a secured microcontroller which handles secret elements (Fig 3).

Open Source or Closed Source: Hardware wallet's software can either be open source or closed source. Open-source software has advantages such that anyone can audit the code,

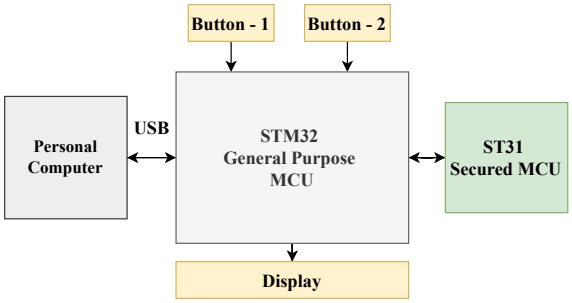


Fig. 2: Hardware Architecture (dual chip design)

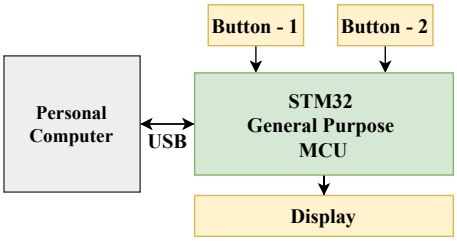


Fig. 3: Hardware Architecture (single chip design)

providing opportunities of improvements but also allowing attackers to exploit weaknesses [12].

### *B. Supply Chain Attack*

Supply chain attack means that device security has been compromised before its delivery to the end user.
Accessibility: The attacker needs physical access to device either during transportation or attacker can buy a device from manufacturer's website, compromising preconfigured encryption and apply for device return as per manufacturer's policy.
Goal: The goal of this attack is to know the seed phrase that user will use in future for cryptocurrency transfer either by malware injection or tempering device encryption chip.
Prevention: Special holographic sticker is used as a proof that device is never tempered [17]. This security check can be compromised without any traceability. Instead, most manufacturer offers device self authentication test on initialization, making sure that encryption chip is in original state [13].

### *C. Hardware Vulnerabilities*

Most critical hardware vulnerability is side channel attack, an exploit that aims to gather system information by measuring hardware emission (power, electromagnetic emanations, acoustic emanations) or influencing program execution by exploiting its hardware (voltage glitching, temperature, clock frequency) [12].
Accessibility: Physical access to hardware is required. Side Channel attacks can be divided into two types:

- **Profiled Attacks**: Attacker must have access of two identical devices. Device A is evaluated, recording its behavior and a data model is constructed, which is then used for device B whose sensitive information is unknown [12].
- **Non-Profiled Attacks**: Attacker has access of only a single closed target device allowing only limited number of side channel traces [18].

Supply Voltage Disturbance: Supply voltage glitch forces hardware to give unpredictable behavior and PIN verification can be bypassed. The attack gives an attacker the opportunity to use the wallet and sign transactions with the wallet. PIN verification by microcontroller takes some time (order of milliseconds). Test is repeated multiple times with different offset (in microseconds) to accurately time the fault injection with critical instruction execution.
Power Consumption Analysis: Encryption algorithm is a combination of mathematical operations (multiplication, addition etc.). Power consumption is different depending on the operation in progress. Precise measurements, multiple test points combined with statistics, algorithm of asymmetric encryption can be estimated and eventually estimating private key [12].
Deep Learning based side channel attacks: Combining side channel attack traces with deep learning metrics, attacker can extract the bits of the keys used in the algorithms [18]. First case study of machine learning in side channel attacks was done in 2011 [19].
Prevention: Hardware Wallet are becoming resistant to external glitching such that device goes to reboot with a warning message but most of the time it is difficult to eliminate. Side channel attacks are tricky to defend.

### *D. Software Vulnerabilities*

Most common software attacks are flawed program design, buffer overflows or application programming interfaces vulnerabilities allowing unauthorized attacker to enter a system. Most common software vulnerabilities are:
**Data Layer Vulnerability:** Hardware wallets have memory protection units allowing APIs to access dedicated memory region [11]. The goal of this attack is to somehow override memory protection unit and gain access of restricted memory region.
Attack: Attacker can have access to device before initialization by user and a known seed is injected into the device. Attacker knows PIN code, installing a rogue API such that can break protected memory isolation.
Case Study: Ledger hardware wallet had a bug causing memory protection unit (MPU) to be misconfigured, allowing attacker to read 16K of memory of supposed to be protected memory region [17], [20]. Similarly, a bug in Trezor hardware wallet allowed write to flash memory [21].
**Communication Layer Vulnerability:** Attacker can extract secret information from device via PC connectivity; commanding device to reveal or manipulate wallet secrets, change PIN even steel secret passphrase.
Attack: Attack is triggered by malicious software on Computer and target device is unlocked either by user for normal operation or by the attacker.
Case Study: This vulnerability was found in Keepkey Wallet [22] and Trezor Wallet such that data sent over USB interface

can trigger a buffer overflow [23] and USB leaked discarded memory [24].

**Application Layer Vulnerabilities:** Attacker can take advantage of flawed program algorithm. Certain portion of device's firmware, bootloader and user data can get accessed during firmware update or rogue application installing.

Attack: Attack is initiated by installing a rogue application or during firmware update.

Case Study: The Ledger kernel had a bug in code; increasing number of system calls incorrectly validated pointer arguments, potentially allowing agents to read data belonging to the kernel or other agents. Also, during firmware update, user sensitive data is copied to RAM. Halting device at this point and with installing a rogue application that can set Debug Flag, contents of RAM are accessible over JTAG [17], [20].

TABLE I. Hardware Wallets Attack Summary

| Attack Type | Goal | Severity | Status | Ref. |
|---|---|---|---|---|
| **Supply Chain** | -Preconfiguration of Seedphrase or PIN | Low | Solved | [17] [13] |
| **Hardware** | -Side Channel Attack (Hardware measurements) -Hardware analysis (Deep Learning based) -Hardware Glitching (Voltage, Clock, Current) | High | Open | [12] [18] [25] [19] |
| **Software** | -Device Memory Access -Code injection to modify encryption -Device flawed algorithm | High | Open | [20] [17] [21] [22] [23] [24] |

## V. Evaluation

Security analysis of hardware wallet is summarized in table I. In the early years of product development, supply chain attacks were profound but in later years, supply chain attacks are becoming less. Hardware vulnerabilities need physical access of the device. Side channel attacks are difficult to detect, tricky to defend and often do not leave any trace. Implementation of deep learning algorithm with side channel attacks have increased attack success rate. Software vulnerabilities need strong knowledge of software architecture and in majority cases, PIN code is also required for attack execution.

## VI. Conclusion

There are two main open challenges in making hardware wallets secure and dependable. First, with the increase in measurement sensitivity, it is possible to gather extremely detailed data about system when running, making side channel vulnerabilities hard to detect at product development stage. Even after its identification, its solution often requires redesigning of hardware architecture which in majority cases is not feasible. Second, there is a need of reviewing cryptographic operations of sending or receiving transactions such that transaction request by attacker can be reverted to the user, making assets recoverable in case of security breach. Hardware wallet security is a process of continuous evolution and further research, and development is required in these two domains.

## References


[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized Business Review*, p. 21260, 2008.

[2] E. G. Julie, J. J. V. Nayahi, and N. Z. Jhanjhi, *Blockchain Technology: Fundamentals, Applications, and Case Studies*. CRC Press, 2020.

[3] M. Arapinis, A. Gkaniatsou, D. Karakostas, and A. Kiayias, "A formal treatment of hardware wallets," in *International Conference on Financial Cryptography and Data Security*, pp. 426–445, Springer, 2019.

[4] H. Rezaeighaleh and C. C. Zou, "New secure approach to backup cryptocurrency wallets," in *2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2019.

[5] M. Huhmo, "Blockchain technology: Bitcoin as a case," 2018.

[6] C. Shaik, "Securing cryptocurrency wallet seed phrase digitally with blind key encryption," *International Journal on Cryptography and Information Security (IJCIS)*, vol. 10, no. 4, 2020.

[7] S. Ahamad, M. Nair, and B. Varghese, "A survey on crypto currencies," in *4th International Conference on Advances in Computer Science, AETACS*, pp. 42–48, Citeseer, 2013.

[8] W. Dai, J. Deng, Q. Wang, C. Cui, D. Zou, and H. Jin, "Sblwt: A secure blockchain lightweight wallet based on trustzone," *IEEE access*, vol. 6, pp. 40638–40648, 2018.

[9] P. Das, S. Faust, and J. Loss, "A formal treatment of deterministic wallets," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 651–668, 2019.

[10] S. Jokić, A. S. Cvetković, S. Adamović, N. Ristić, and P. Spalević, "Comparative analysis of cryptocurrency wallets vs traditional wallets," *Ekonomika*, vol. 65, no. 3, pp. 65–75, 2019.

[11] S. Suratkar, M. Shirole, and S. Bhirud, "Cryptocurrency wallet: A review," in *2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP)*, pp. 1–7, IEEE, 2020.

[12] M. San Pedro, V. Servant, and C. Guillemet, "Side-channel assessment of open source hardware wallets," *Cryptology ePrint Archive*, 2019.

[13] S. Volotikin, "Software attacks on hardware wallets," *Black Hat USA*, 2018.

[14] E. Almutairi and S. Al-Megren, "Usability and security analysis of the keepkey wallet," in *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pp. 149–153, IEEE, 2019.

[15] K. Pfeffer, A. Mai, A. Dabrowski, M. Gusenbauer, P. Schindler, E. Weippl, M. Franz, and K. Krombholz, "On the usability of authenticity checks for hardware security tokens," in *30th USENIX Security Symposium (USENIX Security 21)*, pp. 37–54, 2021.

[16] A. Dabrowski, K. Pfeffer, M. Reichel, A. Mai, E. R. Weippl, and M. Franz, "Better keep cash in your boots-hardware wallets are the new single point of failure," in *Proceedings of the 2021 ACM CCS Workshop on Decentralized Finance and Security*, pp. 1–8, 2021.

[17] R. Team, "Hacking the ultra-secure hardware cryptowallet." https://www.riscure.com/blog/hacking-ultra-secure-hardware-cryptowallet, Aug. 2018.

[18] B. Timon, "Non-profiled deep learning-based side-channel attacks with sensitivity analysis," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 107–131, 2019.

[19] G. Hospodar, B. Gierlichs, E. De Mulder, I. Verbauwhede, and J. Vandewalle, "Machine learning in side-channel analysis: a first study," *Journal of Cryptographic Engineering*, vol. 1, no. 4, pp. 293–302, 2011.

[20] Ledger, "Firmware 1.4: deep dive into three vulnerabilities which have been fixed." https://www.ledger.com/firmware-1-4-deep-dive-security-fixes, Mar. 2018.

[21] Trezor, "Trezor one: Firmware update 1.6.3." https://blog.trezor.io/trezor-one-firmware-update-1-6-3-73894c0506d, Aug. 2018.

[22] InvdBlog, "Security updates in keepkey firmware 7.1.0." https://blog.inhq.net/posts/keepkey-CVE-2021-31616/, Apr. 2021.

[23] Trezor, "Security updates in trezor one firmware 1.6.2." https://blog.trezor.io/details-about-the-security-updates-intrezor-one-firmware-1-6-2-a3b25b668e98, June, 2018.

[24] Trezor, "Security updates in trezor one firmware 1.7.2." https://blog.trezor.io/details-about-the-security-updates-intrezor-one-firmware-1-7-2-3c97adbf121e, Dec. 2018.

[25] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Annual international cryptology conference*, pp. 388–397, Springer, 1999.