Digital Currency Wallet Application Security Audit Report

Loopr Wallet (iOS)

SECBIT

Dec 15, 2018

1. Introduction

Loopr Wallet (iOS) is a digital currency wallet application. SECBIT Labs conducted an audit from August 2018 to September 2018, including an analysis of the wallet (iOS) in 3 areas: **common risks in iOS applications**, **digital wallet assets security** and **server-side application security**. The assessment shows that Loopr Wallet (iOS) has no critical security risks, and SECBIT team has some tips on logical implementation, potential risks and code revising(see part 4 for details).

Type	Description	Level
Assets Security	No anti-screen-capturing	Mid (Fixed in v0.9.10)
Assets Security	No mnemonic regeneration after exiting the page	Mid (Fixed in v0.9.10)
Potential Risk	Private key leakage through system logging after wallet creation	Low (Fixed)

2. Wallet Information

This section describes basic wallet information.

Name	Loopr Wallet (iOS)
System	iOS
Released	No
Source	GitHub
File Type	Source Code
Code Path	https://github.com/Loopring/loopr-ios
Commit id	909c0e94d4b864119663cd2cabe4e544dc87336c
Supported Tokens	ETH, ERC20 Token

3. Wallet Analysis

This section analyzes the functionality and security of the wallet application.

3.1 Functionality Analysis

As a cryptographic digital currency wallet, the implementation of Loopr Wallet (Android) in this area can be divided into 4 primary parts: wallet creation, wallet import, key management and asset transfer.

- Wallet Creation
  - Create one or multiple wallets
  - Users can skip the verification of seed words verification in the wallet creation
- Wallet Import
  - Users can import wallets by seed words, Keystore and private keys
- Key Management
  - Rename wallets
  - Export Keystore files
- Asset Transfer
  - Support sending and receiving ETH and ERC20 tokens

3.2 Security Analysis

- Random Number Generation

  The core part is provided by Keystore library from TrustWallet, which calls the ciphertext generation library from Trezor Wallet and contains no known bugs.

- Wallet Seed Generation

  App applies TrezorCrypto library from Trezor, which uses `srand()` and `rand()` that are marked as unavailable in Swift. We recommend using random numbers generated by `arc4random` for mnemonic generation. Code satisfies BIP39 and no issue has been found.

- Key Derivation

  This function uses functions related to BIP32 of TrustWallet libraries and derives keys according to BIP44. Correct process and coin type. mnemonic words are protected by BIP39 passwords. No incompatibility found.

- Key Storage

  Uses passphrase-protected mnemonic words in BIP39 and derives keys by concatenating mnemonic words and user passphrase. Resistant to brute-force attacks if mnemonic words are leaked.

- Key Management

  Offers multiple key exporting formats.

- Anti-screen-capturing

  No measures found (fixed in v0.9.10).

- Insecure 3rd-party keyboard
  - Not applied
- Network Transmission encryption
  - No data transmitted through network related to secrets, or security-sensitive data
- Sensitive Authority Request
  - No obviously unnecessary authority requests, e.g. phone book, geo-location.

4. Audit Details

This section introduces the audit process and reports issues/risks/tips in detail.

4.1 Audit Process

The audit strictly follows the audit specification of SECBIT Labs. We analyze the project in 3 aspects: the common application security, the asset security and the server security. The audit is processed in following four steps:

- Each audit team reviews the wallet application independently.
- Each audit team evaluates the vulnerabilities and potential risks independently.
- Each audit team shares its audit results, and reviews results from other teams.
- All audit teams coordinates with the audit leader to prepare the final audit report.

4.2 Audit Result

After scanning with SECBIT internal tools and external open-source tools, the auditing team performed a manual assessment. After the inspection into the source code of the wallet application, the results can be categorized into the following types:

Number	Classification	Result
1	Check risks in wallet seed generation and storage	Pass
2	Check risks in private key creation and storage	Pass
3	Check risks in local storage of critical info	Pass
4	Check risks in wallet import	Pass
5	Check risks in wallet password	Pass
6	Check risks in digital currency transfer	Pass
7	Check risks in private key and random number generation algorithm	Pass
8	Check risks in business logic	Pass
9	Check risks in user privileges	Pass
10	Check risks in App runtime environment	Pass
11	Check risks in App development procedure	Pass
12	Check risks in App components	Pass
13	Check risks in App local file and cache storage	Pass
14	Check risks in App and server networking	Pass

4.3 Issues

- No anti-screen-capturing
    - Level: **Mid**

- Type: Assets Security
- Description:

  Users could capture screen when the mnemonic words are displayen on the screen.

- Impact:

  mnemonic words might be leaked when screen captures are revealed or the device is physically attacked.

- Suggestions:

  Refer to practices of conventional Bitcoin wallets:

  - Regenerate mnemonic words if user captures the screen
  - Hide entered words during key importing
- Result:
  - Applied anti-screen-capturing measures, would regenerate mnemonic words every time user captures the screen.

4.4 Risks

Risks are security issues in App running or product design logic. SECBIT team found the following risk after assessing Loopr Wallet (iOS):

- No mnemonic word regeneration after leaving the page
  - Level: **Low**
  - Description:

    Click back button on wallet password entry and mnemonic word display to return to wallet name creation and re-enter wallet name or passwords for next, mnemonic words would not change.

  - Result:

    Would regenerate new mnemonic words every time entering mnemonic word display page.
- No notification for users to remember wallet passphrase
  - Level: **Mid**
  - Description:

The latest Loopr Wallet (UP) uses BIP39 for mnemonic word protection, which differs from common practices. The real key seed is combined by mnemonic words and passphrase so users must both remember seed words and passphrase for private key recovery. If any part is missed, assets would be locked.

5. Conclusion

Loopr Wallet (iOS) implements common wallet functions (wallet creation, wallet import, key management, and asset transfer) according to BIP32, BIP39 and BIP44 with additional functions by specific project targets. SECBIT team had found no critical bug or flaw after analyzing Loopr Wallet (iOS). Some issues and potential risks are found in the wallet application as demonstrated above.

Disclaimer

SECBIT digital wallet audit service assesses the wallet's correctness, security and performability in account security, assets security and potential risks. The report is provided "as is", without any warranties about the code practicability, business model, management system's applicability and anything related to the code adaptation. This audit report is not to be taken as an endorsement of the platform, team, company or investment.

Appendix

Vulnerability/Risk Level Classification

Level	Description
High	Severely damage the digital assets and allow attackers to steal, and/or disable digital assets of users.
Mid	Damage digital assets' security under limited conditions and cause impairment of benefit for stakeholders.
Low	Cause no actual impairment to digital assets.
Info	Relevant to security/best practice or rationality of digital assets and the wallet, not directly imply risks.

SECBIT Labs is devoted to construct a common-consensus, reliable and ordered blockchain economic entity.

https://www.secbit.io

audit@secbit.io

@secbit_io