Hiwallet 安全性

关于 Hiwallet 的安全主要的关注是在用户的钱包信息是否会泄露。Hiwallet 把钱包的数据存储在 `加密数据库` 中，也就是我们要关注 `加密数据库` 是否会存在泄漏。

加密数据库

Hiwallet 数据库分为两类： `加密数据库` 和 `不加密数据库` ， `加密数据库` 的加密方式是对数据库加密码，但不对存入数据库中的数据加密。 `加密数据库` 的密码由由两部分组成：128 位随机熵（随机 `bit` ）部分 + `Pin 码` 加密部分 ，下面对这两个部分内容做一些说明：

> 数据库密码 = 128 bit 的随机数 + `Pin 码` 的一次 `sha512`

- 128 位随机熵部分（下面使用 `随机熵` ）

  用户在注册时输入的 `Pin 码` （6 位 [0 ~ 9] 的数字）之后， `Hiwallet` 会纯随机 128 bit（16 Bytes）的一个数据，然后把这个数据存储到 Keychain 中。理论上暴力破解 `随机熵` 的概率是极低，哪怕用 1 T/s 的计算机也要万年以上的计算才可破解。

- `Pin 码` 加密部分

  我们在用户在注册时输入的 `Pin 码` （6 位 [0 ~ 9] 的数字）之后，将 `Pin 码` 进行一次 `sha512` 操作后作为打开加密数据库的密码的一部分；同时把 `Pin 码` 的两次 `sha512` 存入非加密数据库内，作为每次使用 `Pin 码` 登陆时的验证密码。

Keychain

`Hiwallet` 有两种登陆方式： `Pin 码` 方式和生物识别（ `指纹` 或者 `Face ID` ）方式. 根据上面的 `加密数据库` 的密码规则，我们存储在 Keychain 的内容也是不一样的。

> 数据库密码 = `随机熵` + `Pin 码` 的一次 `sha512`

- `Pin 码` 方式

  由密码规则知道： `Pin 码` 可由用户输入得到，上面知道 `随机熵` 存储在 Keychain 中。这样当用户输入正确的 `Pin 码` 后即可打开数据库。

- 生物识别（ `指纹` 或者 `Face ID` ）方式

  用户也可以使用生物识别（ `指纹` 或者 `Face ID` ）方式登陆，这种方式用户不输入 `Pin 码` ，那么 `加密数据库` 怎么打开呢？针对着这种方式，我们会在开启生物识别时候把 `Pin 码` 存入到 Keychain 中，也就是说除非本人打开（生物信息复制或丢失依赖硬件设备，暂不考虑该情况），否者拿不到 `Pin 码` 和 `随机熵` 。

iOS 用的加密库

- `CryptoSwift`
  - `sha512`
- Security
  - public func SecRandomCopyBytes(_ rnd: SecRandomRef?, _ count: Int, _ bytes: UnsafeMutableRawPointer) -> Int32 （用来纯随机）
- `KeychainAccess`

审计需要关注的问题

1. 这种只加密数据库的方式是否会有漏洞?
2. 数据库密码（ `随机熵` + `Pin 码` 的一次 `sha512` ）设计是否合理?
3. 数据库密码是否存在丢失或者破解的风险?