2020-09-01 19:34

**一文读懂 WalletConnect**

[[](https://www.8btc.com/writer/114510)TODD](https://www.8btc.com/writer/114510)发布在[链圈子 海盗号](javascript:;)

 141.0万



**文章概览**

* WalletConnect 能干什么
* WalletConnect 的市场支持情况
* WalletConnect 消息通讯工作原理
* WalletConnect 消息安全如何保障
* WalletConnect 如何快速接入推送

**01 WalletConnect 能干什么**

它是⼀个基于⼆维码建立连接的基础通讯设施。你可以在它的基础上去建⽴各种交互，它默认的交互协议都是与以太坊的交互。当然你也可以做能够想象到的各种事，可以不只局限于 Dapp 通讯交互，如果你把默认的以太坊协议改掉，也意味着你将与其他⽀持 WalletConnect 钱包、Dapp 脱离关系，互不兼容。

现在出现了很多针对电脑这种大屏幕的 Dapp 应用，在 WalletConnect 没有出现之前，必须使用Chrome 插件或者桌面端钱包来授权操作，这让原本使用移动端钱包的用户操作和使用习惯出现了割裂感，还需另外操作下载插件或钱包，再将私钥导⼊进去，很麻烦。

**02 WalletConnect 的市场支持情况**

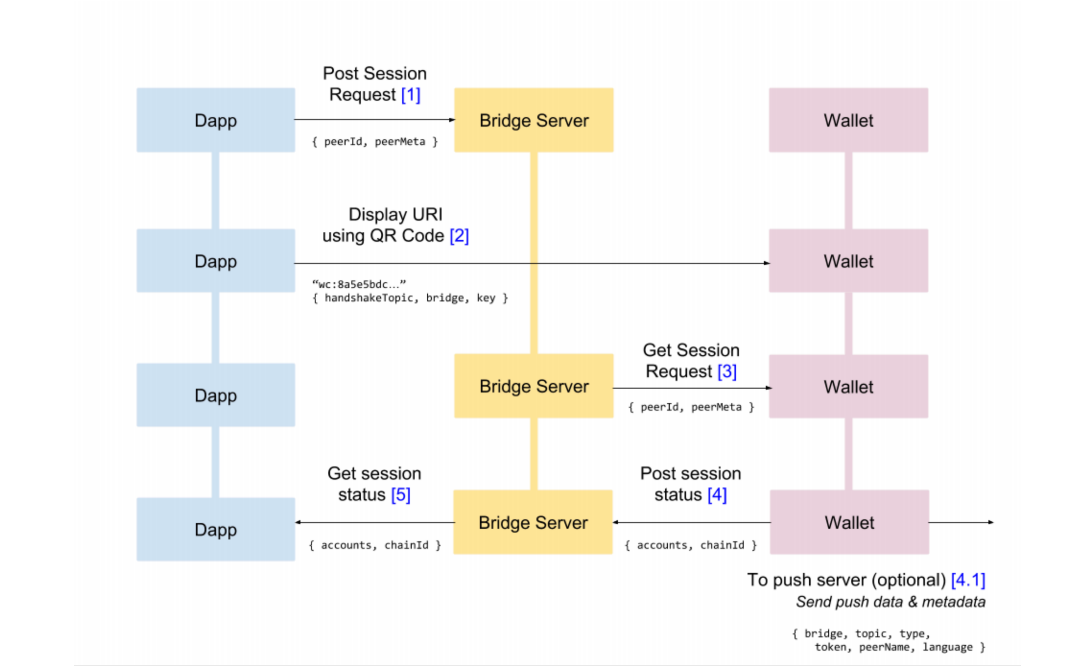
最近较为常见的钱包都在支持，比如:

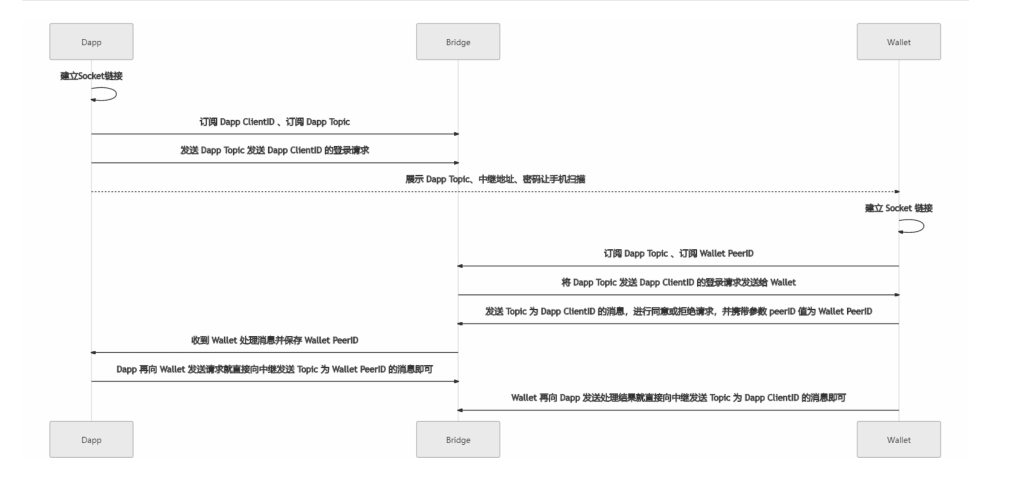
* imToken
* TokenPocket
* 麦子钱包
* Trust Wallet

**03 WalletConnect 消息通讯工作原理**

**//**WalletConnect 建立连接原理

Topic 名词解释：文本意思为“主题”，在 WalletConnect 中钱包端与 Dapp 端各有⼀个 Topic 主题，让对方订阅，这样就形成了⼀个可以相互通信的通道。





1. Dapp 端与中继服务器建立 Socket 连接；
2. Dapp 端⽣成 Dapp ClientID（监听该 Topic 会得到发送给 Dapp 的消息），Dapp Topic（监听该Topic 会得到 Dapp 发出的登陆请求），并订阅 Dapp ClientID；
3. Dapp 端向中继服务器发送 Topic 为 Dapp Topic 的信息，并携带 Dapp ClientID 信息；
4. Dapp 端展示⼆维码包含 Dapp Topic 信息，中继服务器的地址，密码；
5. 钱包端扫描 Dapp 端的⼆维码，解析出⼆维码信息获得 Dapp Topic，中继服务器地址，密码；
6. 钱包端与中继服务器建立socket 连接，然后产生 Wallet PeerID（监听该 Topic 可以获得发送给钱包端的消息），并订阅 Dapp Topic 和 Wallet PeerID ；
7. 接下来钱包端将接收到中继服务器转发过来的 Dapp 的登陆请求；
8. 钱包端处理是否同意 Dapp 的登陆，发送 Topic 为 Dapp ClientID 的消息，并把处理结果和Wallet PeerID 的信息传递回去；
9. 此时连接建立成功，如果「Dapp」想给「钱包端」发送消息，则发送 Topic 为 Wallet PeerID 的信息即可；
10. 如果「钱包端」想给「Dapp」发送消息，则发送 Topic 为 Dapp ClientID 的信息即可。

钱包端 Topic 和 Dapp 端 Topic 两方共同组成了 Session。

WalletConnect 的 Session 生命周期才是 Dapp 与钱包双方建立的连接的生命周期，WebSocket 只是它的“通讯⼯具”，WebSocket 的断开不代表此次与钱包的连接断开。

**04 WalletConnect 通讯工作原理**

* WalletConnect 只是⼀套通讯协议，双⽅通信什么信息都可以；
* WalletConnect 默认通过 WebHook 的⽅式来⽀持推送服务，你可以在⼿机 App 没有打开的情况下来通知⽤户处理信息；
* 如果有⼀方 A 断开 Socket 连接，另⼀⽅ B 发送消息会暂存在中继服务中，当下⼀次 A 与中继服务器建⽴ Socket 连接，并订阅相关的 Topic，才会将中继服务暂存的消息发送给 A。

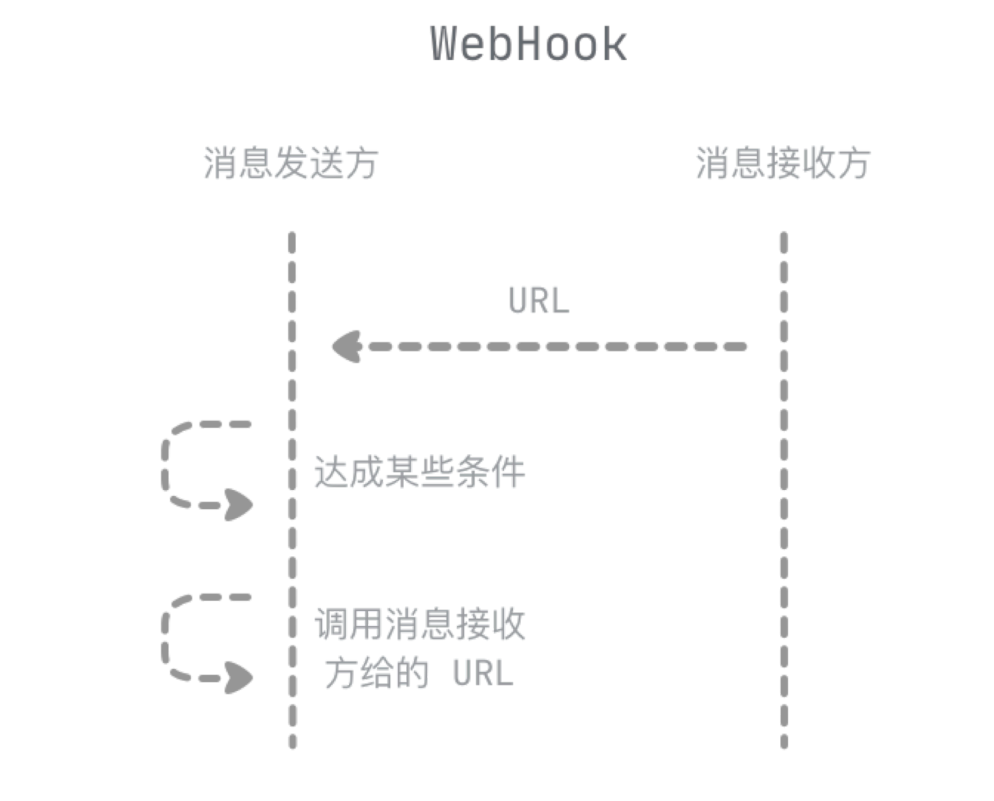
**05 WalletConnect 消息安全如何保障**

1. 使用 AES-256 对称加密来加密通讯的信息，HMAC-SHA256 做 Hash 签名；
2. Dapp 端⽣成的⼆维码中包含对称加密的密码，⼿机端扫描该⼆维码获得密码，所以 Dapp ⼆维码不会过网传递。

**06 WalletConnect 如何快速接入推送**

**//**底层做支持的技术 WebHook

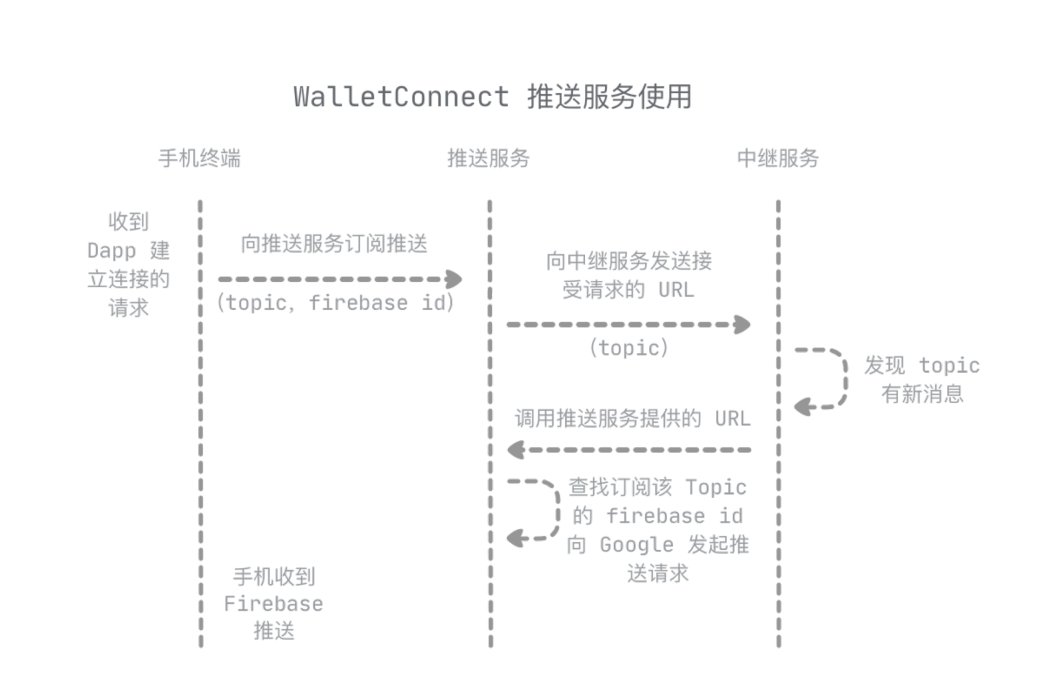
* WebHook 是⼀种 HTTP 回调：某些条件下触发的 HTTP POST 请求；
* WalletConnect 的中继服务中也有这个接受 WebHook URL 的 API 接⼝；
* WalletConnect 单独提供了 Firebase 推送服务。



1. 消息接收方将⼀个 URL 和想接收的消息发送给消息发送⽅；
2. 消息发送方达成⼀定的条件，就会去调⽤ 消息接收⽅的 URL，来通知消息接收方接收消息。

**//**WalletConnect 推送使⽤流程

1. ⼿机端收到 Dapp 的登陆请求，⼿机端如果同意登陆请求，可以使⽤ Topic Dapp ClientID 和Firebase ID 向「推送服务」订阅消息；
2. 「推送服务」收到 Topic 会继续向「中继服务」发送 WebHook 订阅 Topic；
3. 如果「中继服务器」发现「推送服务」订阅的 Topic 有信息，它就会通知「推送服务」；
4. 「推送服务」找到订阅相关 Topic 的 fitebase ID 发起推送。



**07 总结**

WalletConnect 是⼀种通过扫描 QR 码使⽤端到端加密技术将桌⾯ DApp 连接到移动设备的开放性协议。它开启了整个曾经只有Metamask 钱包可⽤的 DApps 世界。⽤户可以在不损害其私钥安全的情况下与任何 DApp 交互，并能在其移动设备上收到通知，签字同意任何交易请求。