与「十」俱进，码出未来

2018源创会年终盛典

主办方 开源中国 oschina.net

Content

去中心化和中心化在安全防控方面的区别

去中心化应用主要的安全隐患

去中心化应用安全解决方案

ONTology

去中心化应用安全防控特征

- 机制公开透明

去中心化应用秉承了区块链开放精神，在实现原理、协议、代码实现上均最大化开放，确保其算法和实现原理的公开透明，同时对安全也提出了非常高的要求。

- 充分利用密码学

不同于中心化系统中的防护手段（比如异构的防火墙、专有网络），在去中心化应用中，无法使用这些封闭手段，而是运用了大量的最底层密码学来解决安全问题，包括多重签名算法、同态加密、HASH算法、多方安全计算等等。

- 安全责任终端化

DAPP把安全防护手段更多地推向了终端用户 ，终端用户需要为自己的安全担负全部或者部分责任。比如用户需要理解什么是私钥，如何妥善保管自己的私钥。

去中心化应用的安全隐患

去中心化应用的安全隐患

引自白帽汇在《区块链产业安全分析报告》中的阐述

以上安全事件中，包括区块链自身安全、智能合约安全、应用平台安全（DAPP）、用户自身安全等四种类型。 区块链自身安全和智能合约安全属于区块链安全范畴，不在今天话题中，今天重点关注解决应用平台安全。

ONTology

用户行为安全

引自白帽汇在《区块链产业安全分析报告》中的阐述私钥是区块链交易发起的关键保证，私钥的安全成为DAPP的防护重点，在典型的DAPP中，私钥为用户保管，用户对私钥的保护能力成为DAPP最大的安全隐患。列举2018年典型安全事件如下：

2018年3月	印度女子与骗子共享证书，比特币被盗	受害者损失约 5.5 万美元的比特币
2018年2月	有人在 twitter 上冒充知名人士如 Vitalik Buterin 诈骗数字货币	诈骗分子在短时间内诈骗到约20000 美元的数字货币
2018年1月	IOTA 社区大量用户钱包代币被窃取	给用户造成大量损失

DAPP应用安全

- 服务商服务器被黑客攻击

每年都有大量的交易所被黑客攻击，导致数字资产大量丢失。由于加密数字货币的不可追溯、不可挂失、匿名的特性，一旦数字资产丢失，就再也不可能找回。

- 账户名和密码的安全隐患

大多数用户习惯性的使用同一个账户名和密码在不同的网站里面进行注册，如果这其中有任意一个网站被黑客攻击成功，那么这个网站的全部用户名和密码都可能被用于对交易所账户系统的登录攻击。黑客在获取对于用户托管账户的控制权限后，即可对数字资产进行转移。

- 内部操作员窃取私钥

部分DAPP应用内部管理水平很低，也没任何监管要求，导致内部人员窃取私钥问题防不甚防。

2018年4月	印度比特币交易所Coinsecure公司钱包遭窃取	Coinsecure 被盗取 438 比特币，价值超过 300 万美元
2018年3月	虚拟货币 Verge(XVG)，遭到攻击	黑客窃取了价值 100 万美元的XVG 代币
2018年3月	迪拜某加密货币交易所员工窃取 20 万美元的加密货币供个人使用	交易平台损失价值约 20 万美元的加密货币

运行环境的安全风险

私钥是存储在终端设备上的，无论是PC端还是移动端，终端设备如果出现不安全的现象，对于私钥来说是有非常高的安全风险的。黑客利用操作系统漏洞，可以轻易的绕过操作系统设计的一系列安全边界或沙箱机制，获得访问加密数字货币私钥的能力。无论是Android、iOS、Windows 还是 Linux，每年都有大量的安全漏洞被公开和被修复，而这些漏洞里面就有不少本地提权的漏洞。利用这些提权漏洞就可以轻易的打破操作系统的安全设计边界，获得访问应用的私钥/助记词的能力。

密码算法的安全性

目前区块链基于的算法主要是公钥算法和哈希算法，其安全性来源于数学难度，相对是安全的。但是随着高性能计算和量子计算的发展和商业化，目前所有的加密算法均存在被破解的可能性，这也是区块链面临的一个威胁。 例如MD5 和 sha1 摘要算法，目前已经证明安全性不足，现在已经不能被商用。

去中心化应用安全解决方案

ONTology

私钥的KEYSTORE存储方案

私钥是去中心化应用的安全核心，需要严格保护私钥安全。KEYSTORE设计需要解决几个关键问题，

第一，需要确保私钥被加密存储；

第二，私钥不能被暴力破解；

第三，防止黑客通过字典式攻击，比对出常用口令。

私钥的KEYSTORE存储方案

1 Scrypt算法

2016年，Scrypt算法已经被国际互联网工程任务组IETF纳入RFC 7914规范

Scrypt算法基本过程是，利用一个密钥延展函数（KDF）将口令延展成一个对称密钥，然后用该密钥加密私钥，

- 密钥延展函数是基于Scrypt函数的，该函数的突出特点是，需要耗费大量的内存，使得黑客很难并行地去枚举；

- 密钥延展函数的计算时间是可调节的，当该函数被配置成计算耗费CPU计算时间模式时，黑客的平均试错时间也就大大增加了，从而减慢其试错次数，降低穷举成功率。

```
{
  "n": 16384,
  "r": 8,
  "p": 8,
  "dkLen" : 64
}
```

`n` is a parameter that defines the CPU/memory cost. Must be a value 2^N.

`r` is a tuning parameter.

`p` is a tuning parameter (parallelization parameter). A large value of p can increase computational cost of SCrypt without increasing the memory usage.

`dkLen` is intended output length in octets of the derived key.

2 加盐Salt处理

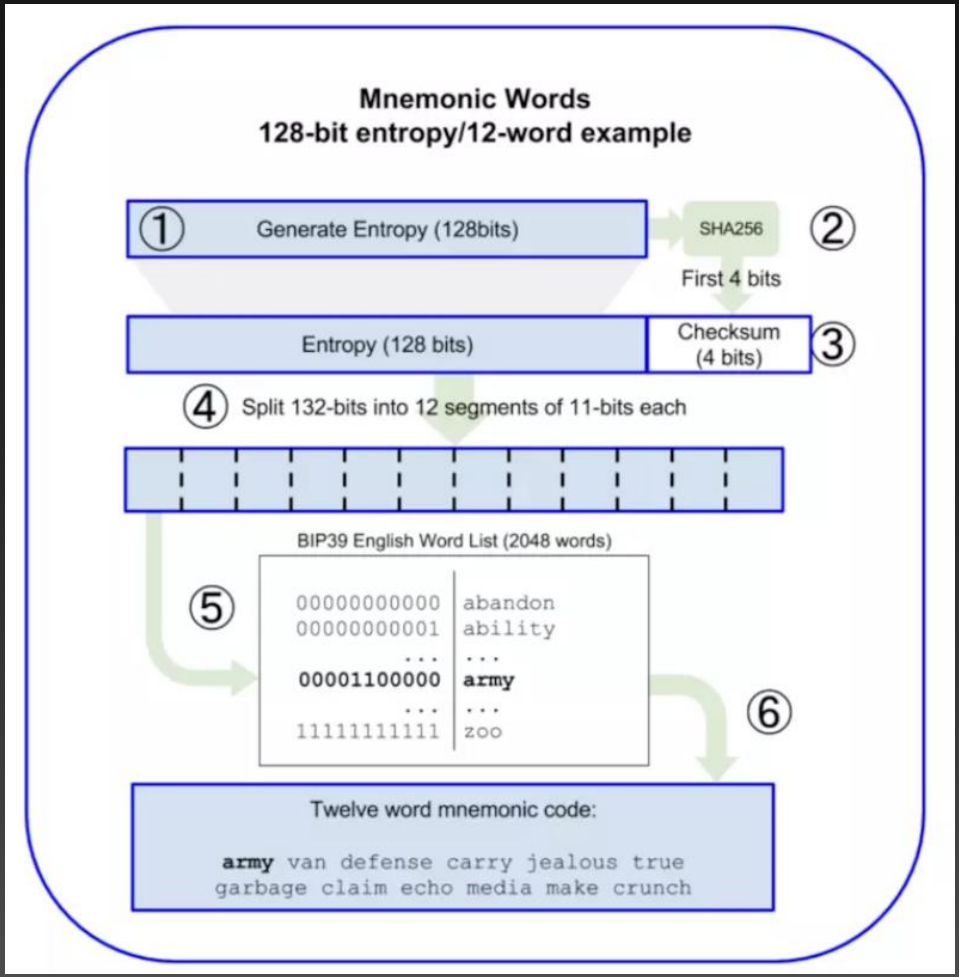
为了增加攻击者的难度，防止攻击者通过查表方式对比口令。对于每一个私钥生成，随机产生随机数，称为Salt值，与用户口令拼接一起进行加密

开放标准：https://dev-docs.ont.io/#/docs-cn/SDKs/01-wallet-file-specification

ONTology

私钥的管理方案

为了保证资产的安全，去中心化应用需要为用户提供除了KEYSTORE以外的异构方案，组合成一个双备份方案，确保用户KEYSTORE丢失后，仍旧有一个恢复介质。这里不得不提的是助记词方案。提出助记词最著名的开放协议是：BIP-39



ONTology

多重签名方案

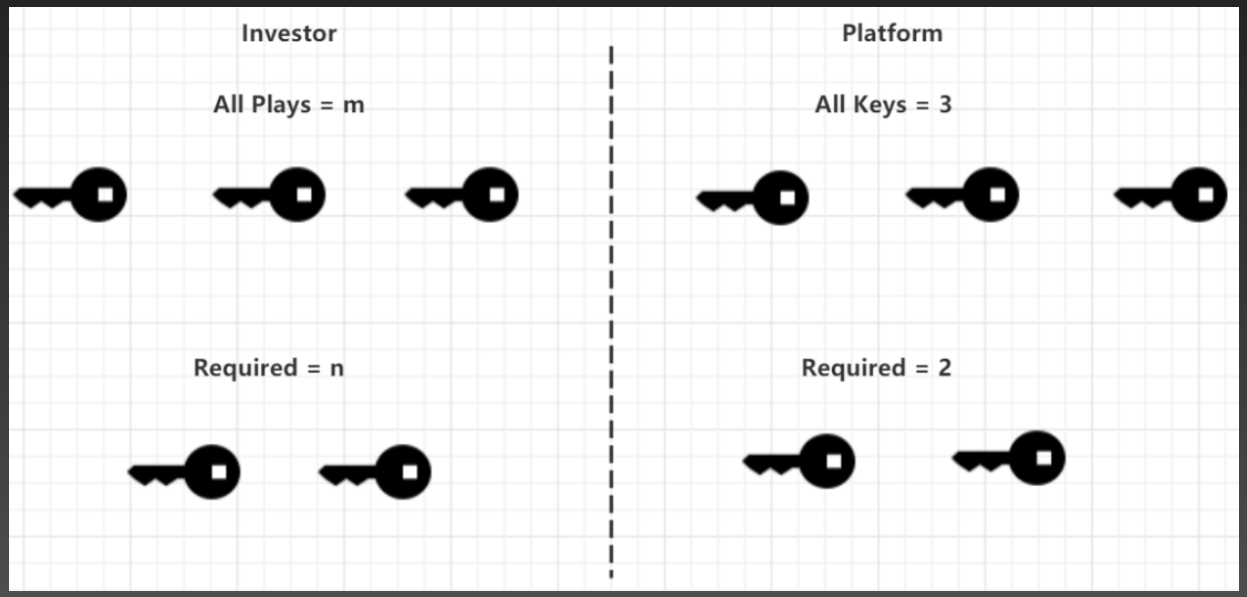
多重签名就是多个用户对同一个消息进行数字签名，可以简单理解为一个数字资产多个签名。

多重签名的作用意义非常明显：

1. 分散对交易/数据/资产的控制权，适合机构或公司的管理模式，比如业务和财务分离管理；

2. 提供密钥冗余和灵活管理机制，比如2/3签名表示，3个人中的任意2人可以完成签名；有效避免单点故障问题。

3. 应用内部系统分散密钥，确保应用内部运营安全。



ONTology

门限签名方案

在现实应用中，基于多重签名方案仍旧存在一些不足，我们会遇到以下问题：

1. 不是所有区块链都能支持多签验证逻辑，比如Ripple网络无法直接支持多签合约；
2. 多签账户所包含的子账户列表是对所有人公开，可能会造成一些隐私的丢失和安全性风险；

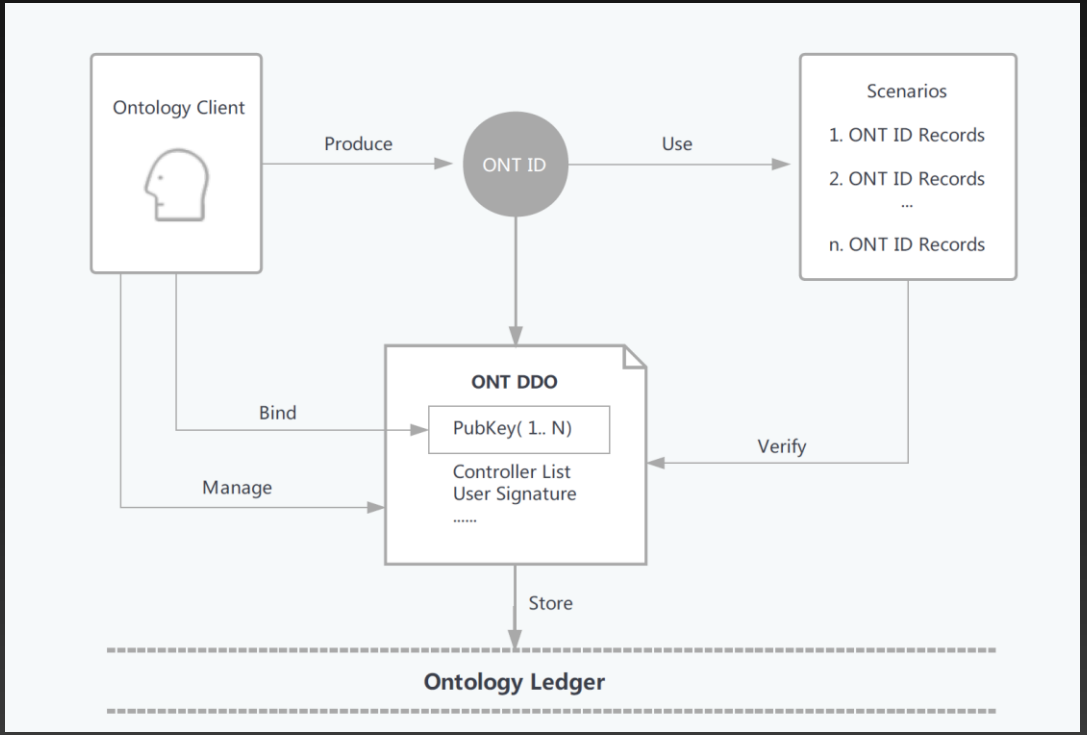
通过门限签名方案能解决以下几个问题：

1. 一个算法的实现就可以支持多种区块链底层，如Bitcoin、Ethereum、Ontology、Ripple等等；
2. 能够保护多签账户的所有参与者的隐私。

去中心化的身份协议ONT ID

- ONT ID是一个去中心化的身份标识协议，ONT ID具有去中心化、自主管理、隐私保护、安全易用等特点。
- 用户自主掌控密钥和对ID的控制权
- 完全兼容 W3C DID 国际标准
- 多密钥、多算法支持
- 身份丢失可恢复

ONT ID的所有者可以设置恢复人代替本人行使对ONT ID的管理权，如修改ONT ID对应的属性信息，在密钥丢失时替换密钥。恢复人可以实现多种访问控制逻辑，如“与”、“或”、“(m, n)-门限”

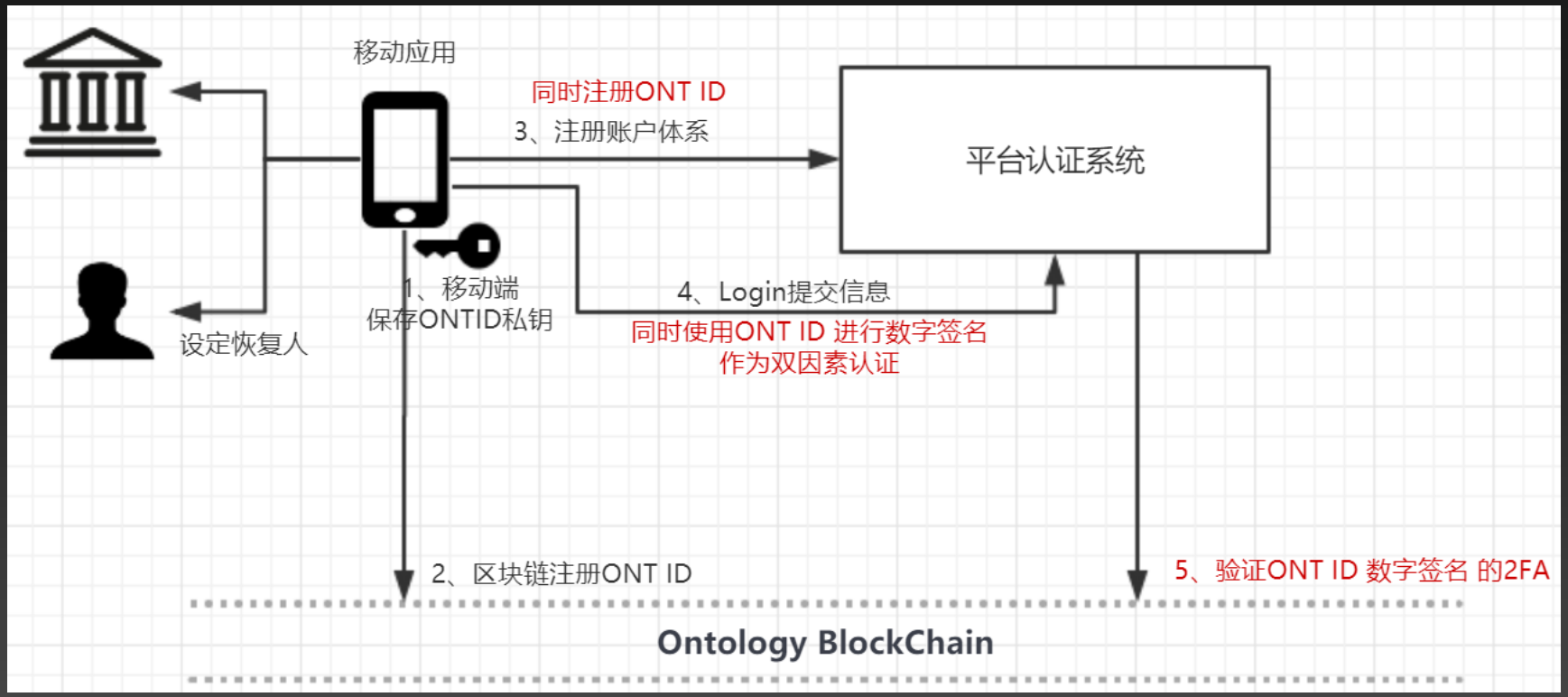


开放标准：https://github.com/ontio/ontology-DID/blob/master/docs/cn/ONTID_protocol_spec_cn.md

ONTology

开放的2FA解决方案

基于ONT ID 开放双因素认证解决方案



ONTology

End-to-End 加密

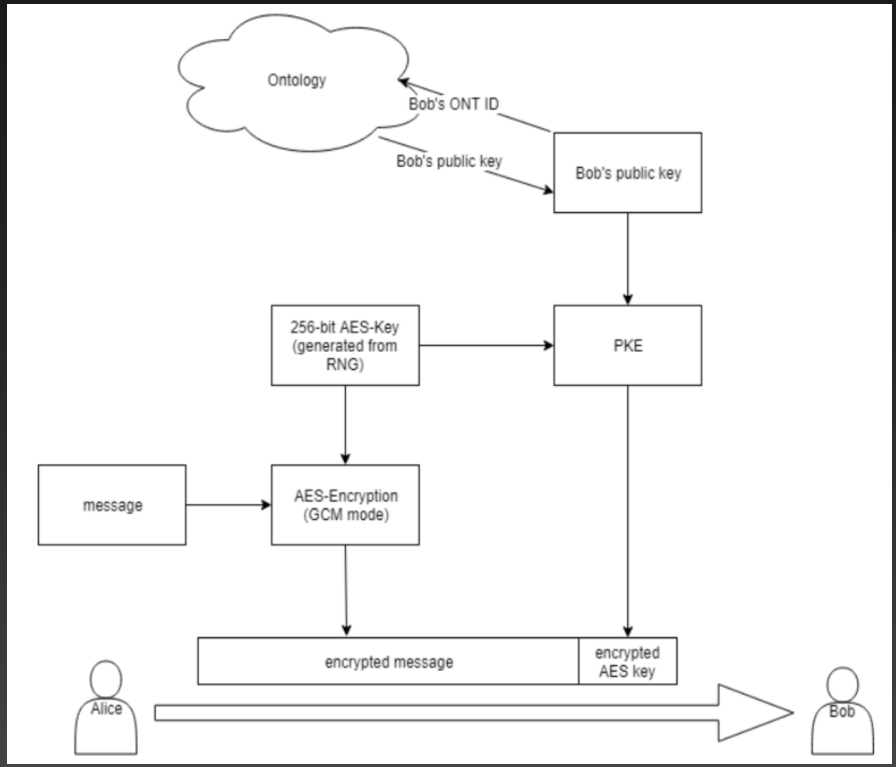
基于ONT ID，我们设计一套混合加密机制，实现端到端加密，避免中间人攻击（Man-in-the-MiddleAttack）。

具体流程包含三个步骤：

1. 获取公钥：访问 Ontology 区块链，获取对方的公钥列表，并从中随机选择一 个公钥 pk；

2. 随机采样 AES 密钥：随机采样 256 比特数据，作为 AES 分组密码的密钥 secret key；

3. 混合加密： 将 256-bit AES 密钥用公钥加密算法进行加密得到 encryptedKey，数据使用 AES 算法 GCM 模式进行加密得到 ciphertext = AES-256-GCM(secret, IV, AAD, data)。



开放标准：https://github.com/ontio/ontology-DID/blob/master/docs/cn/end-to-end-encryption.md

ONTology

前端页面安全

Web 前端的安全主要有三类：XSS、CSRF、界面操作劫持。

XSS（Cross Site Scripting），即跨站脚本攻击，是一种代码注入攻击，恶意攻击者往 Web 页面里插入恶意 Script 代码，当用户浏览该页之时，嵌入其中的 Script 代码会被执行，从而达到恶意攻击用户的目的。

CSRF（Cross Site Request Forgery），即跨站请求伪造，通过伪装来自受信任用户的请求来利用受信任的网站。

界面劫持操作是一种基于视觉欺骗的Web会话劫持攻击，包括点击劫持、拖放劫持和触屏劫持三种类型。

Http头部的安全措施：

- Content Security Policy 可以设置网页可以使用的资源，是个白名单设置。

- Frameguard to prevent clickjacking 可以确保自己网站的内容没有被嵌到别人的网站中去，也从而避免了点击劫持（clickjacking）的攻击。

- Strict-Transport-Security 确保使用了https。

- noCache 可以确保浏览器不缓存文件，保证用户使用最新版本。但是服务端一旦更新文件，应该在文件名上增加一个_x0008_hash，确保文件更新，这个设置不是必须的。

- set X-Content-Type-Options header to nosniff 可以防止浏览器猜测文件内容，从而执行一段脚本。

- X-XSS-Protection 可以预防一种简单的XSS攻击。

移动端安全加固

安全加固的核心功能在于防止各类针对移动应用恶意攻击行为。

（1）反编译保护：通过整体加密、类抽取、VMP保护等技术，防止第三方通过反编译、内存dump等方式获取到程序核心源代码。

（2）防调试保护：同过双向ptrace防调、实时查看、inotify监控等技术，防止第三方通过控制、调试等攻击方式获取关键业务逻辑。

（3）完整性保护：通过签名校验、随机交叉校验等技术，防止第三方盗用公司核心源代码，防止通过二次打包、篡改等方式恶意抹黑公司、植入第三方广告等行为。

Q & A