零知识证明及其当前应用

零知识证明（Zero-Knowledge Proof或Zero-Knowledge Protocol）是一种基于概率的验证方法，它包括“类似事实的陈述”和“关于个人知识的陈述”。

验证者基于一定的随机性来询问证明者，如果证明者给出的答案正确，那么证明者将有很大概率会拥有其所声称的“知识”。**零知识证明可以在不透露使用哪种货币的情况下验证你确实花了钱。**

如今，零知识证明已经被许多区块链项目视为最好的隐私保护方案之一。能够在不泄露数据的情况下，来证明数据的真实性。

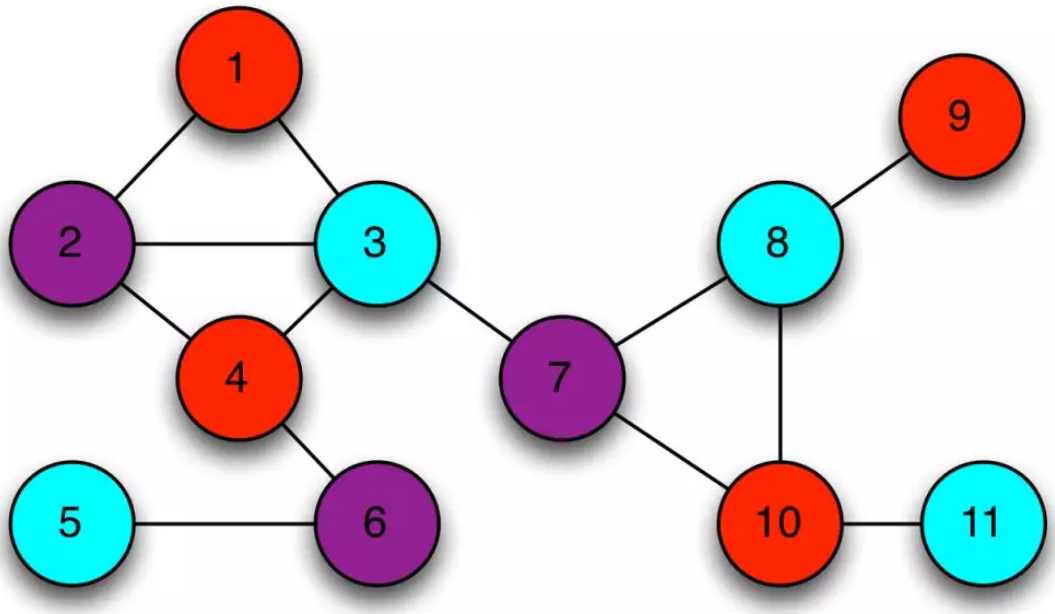
在本文中，我们将会解释神秘的零知识证明（Zero-Knowledge Proof）及其当前的应用。

为什么ZKP如此重要?

数据隐私是当今社会最重要的课题之一。保护与个人身份有关的个人数据（出生日期、银行证明、交易记录、教育证书）是至关重要的，并且其重要性将不断提高。

在科技时代，我们正在创造着前所未有的海量数据，而且我们也在不断创造着关于自身的可供掠夺的数据。

像Google和Facebook这样的大公司利用我们的数据成为了当今世界的科技巨头。然而，最近密码学的突破和区块链的兴起为保护我们的数据和身份信息提供了新方法。零知识证明或许就是答案。



零知识证明的原理

零知识证明是麻省理工学院的研究人员在20世纪80年代提出的一种加密方案。**零知识证明协议是指一方（证明方）可以证明某事对另一方（验证方）来说是真实的。**除了此特定陈述属实之外，不会透露其他任何信息。

例如，当前网站将用户密码的Hash散列值储存在其web服务器中。为了验证客户端是否真的知道密码，大多数网站目前使用的方法是要求客户端输入密码的hash散列，并将其与储存的结果进行比较。

零知识证明可以保护用户账号不被泄漏。**如果可以实现零知识证明，那么客户端密码对任何人来说都是未知的，但是仍然可以对客户端登录进行身份验证。**当服务器受到攻击时，用户的账户仍然是安全的，因为其密码并没有被储存在web服务器中。

零知识证明可以分为「交互式」和「非交互式」两种。接下来我们就一一来看看这两种证明方式有哪些不同。

交互式零知识证明

零知识证明协议的基础是交互式的。它要求验证者不断对证明者所拥有的"知识"进行一系列提问。

例如，如果有人声称自己知道数独游戏的答案，零知识证明的过程就是验证者需要随机指定要通过列、行或九个正方形进行验证。

每轮测试不需要知道具体的答案，只需要检测数字1~9是否包含在内。只要验证的次数足够多，就有理由相信证明者是知道数独问题答案的。

然而，这种简单的方法并不能使人相信证明者和验证者都是真实的。在数独这种情况下，两者可以提前串通，以便证明者可以在不知道答案的情况下依然通过验证。

如果他们想要说服第三方，验证者还必须要证明验证过程是随机的，并且他不会向证明者泄漏答案。

因此，第三方难以验证交互式零知识证明的结果，要向多人证明某些东西的话则需要额外的努力和成本才行。

非交互式零知识证明

顾名思义，非交互式零知识证明不需要交互过程，避免了串通的可能性，但是可能需要额外的机器和程序来确定实验的顺序。

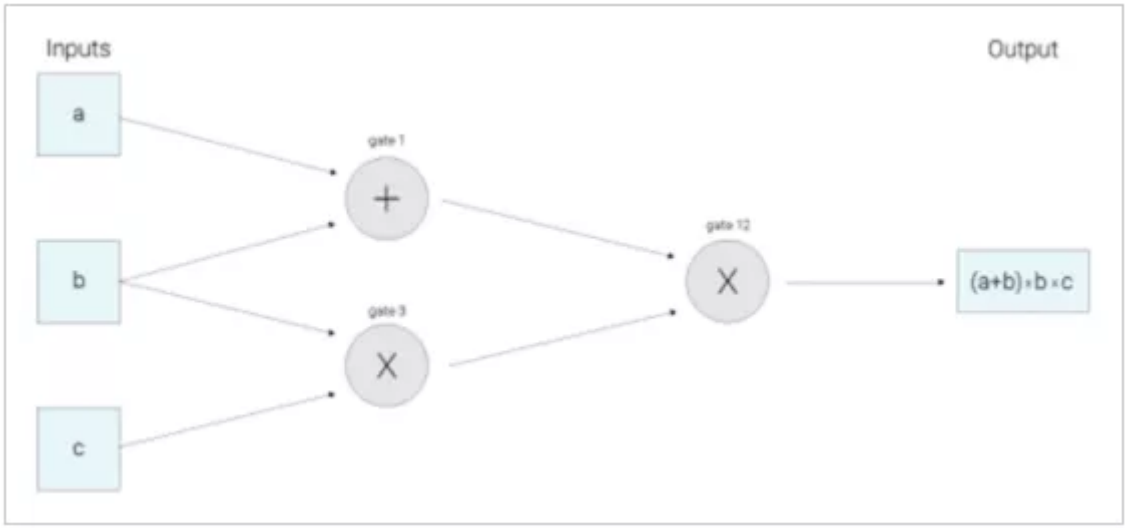
例如，在数独这个例子中，由程序决定要验证的列或行。验证序列必须保密，否则验证者可能会在不知道真正"知识"的情况下通过验证。

区块链上的零知识证明

比特币和以太坊都是用公共地址来代替当事方的真实身份，使交易部分匿名，公众只知道发送和接收的地址以及交易的数量。**然而，可以通过区块链上可用的各种信息（如交互记录）找出地址的真实身份，因此存在暴露隐私的可能。**

零知识证明，可以在发送方、接收方和其他交易细节都保持匿名的情况下，保证交易有效。

ZCash可能是成功实现零知识证明的最著名的区块链项目之一。Zcash实现了ZKP的修改版本，被称为zk-SNARKS，代表Zero-Knowledge Succinct Non-Interactive Argument of Knowledge（零知识简明非交互式知识证明）



zk-SNARK技术减少了证明的大小以及验证所需的计算量。它能够在不泄漏有关地址和相关有价值的任何关键信息的情况下证明有效交易条件得到了满足。

zk-SNARK将需要验证的交易内容转换为两个多项式乘积相等的证明，并结合同态加密和其他先进技术，在执行交易验证时保护隐藏的交易金额。

其过程可以简单地描述为：

- 将代码拆分为可验证的逻辑验证步骤，然后将这些步骤拆分为一个由加法、减法、乘法和除法组成的运算电路；
- 进行一系列变换，将待验证的代码转化为多项式方程，如t(x)h(x)= w(x)v(x)；
- 为了使证明更加简洁，验证者预先随机选择几个检查点s 来检查这些点的方程是否为真；
- 通过同态编码/加密，验证者在计算方程式时不知道实际输入值，但仍然可以进行验证；
- 在方程的左边和右边，同时乘以一个不等于0的秘密值k。当验证(t(s)h(s)k) 等于(w(s)v(s)k)时，具体的t(s)、 h(s)、 w(s)和v(s) 是不可知的，从而达到保护信息的目的。

但zk-SNARK并不是完美的。**当前zk-SNARK实现中的一个缺陷，是需要提前设置参数。****如果这些参数被泄漏，那么整个网络将面临毁灭性的打击**。因此，在使用这些网络时，用户必须坚信参数不会被泄漏。

可能的解决方案包括使用现代"可信执行环境"，如因特尔 SGX以及ARM TrustZone。对于因特尔的SGX技术来说，即使应用程序、操作系统、BIOS或VMM受到威胁，私钥也是安全的。

此外，最近的一份白皮书揭示了它在零知识密码学方面的创新：ZK-STARKs (零知识可扩展透明知识理论，Zero-Knowledge Scalable Transparent ARguments of Knowledge)。

根据zk-STARK 白皮书，zk-STARK是第一个在不依赖任何信任设置的情况下实现区块链验证的系统，随着计算数据的增加，计算速度呈指数级增加。

它不依赖于公钥加密系统，更简单的假设使其在理论上更加安全，因为它唯一的加密假设是Hash 散列函数（例如SHA2）是不可预测的。

不可否认的是，零知识证明和zk-S(T|N)ARK技术的测试和采用都将需要一定的时间。但是对于区块链底层开发平台来说，如何兼顾性能和安全性是至关重要的。也许只有零知识证明等密码学技术被更多应用的时候，区块链这项技术才能被更好地推动。