EDU

CSDN学院　IT实战派

比特币和区块链的相关技术细节

P2P网络和数字签名

讲师：康烁

本章目标

1. P2P网络的简介和分类
2. 数字签名的原理
3. 哈希函数的原理
4. 哈希指针的原理和作用
5. 比特币的账户和交易
6. 共识算法和区块链的关系
7. 不同共识算法的比较

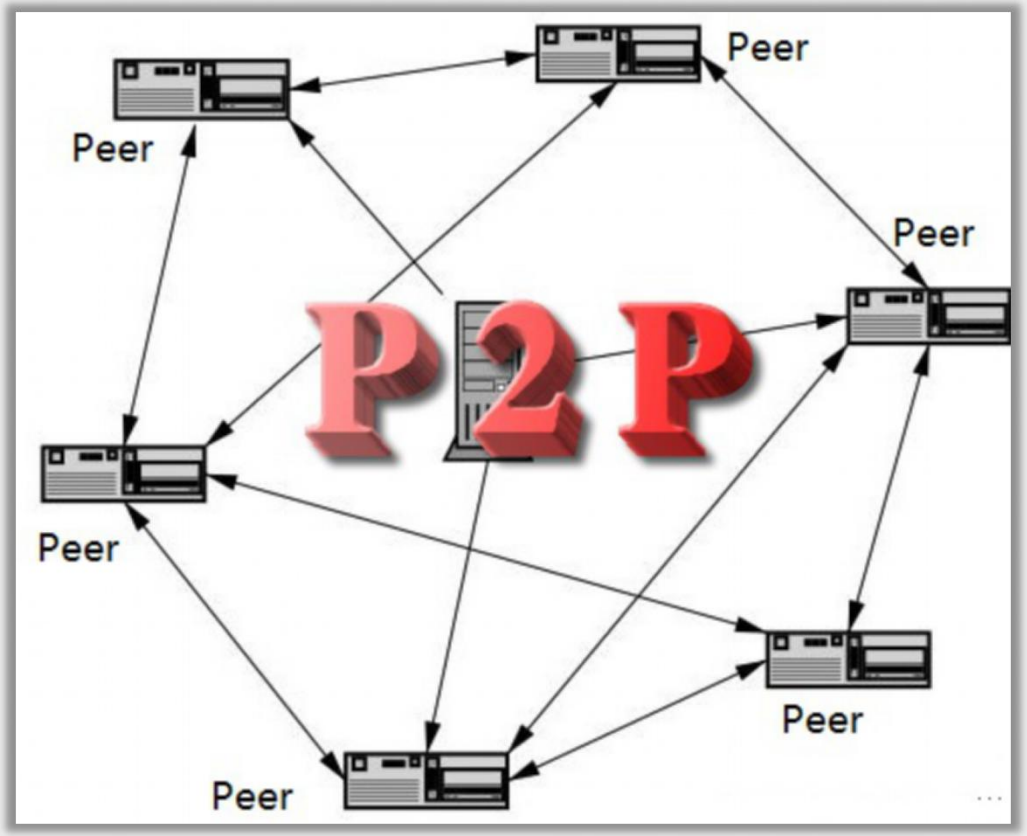
本节目标

1. 了解P2P网络的起源和分类
2. 理解非对称加密的原理
3. 理解数字签名的基本概念

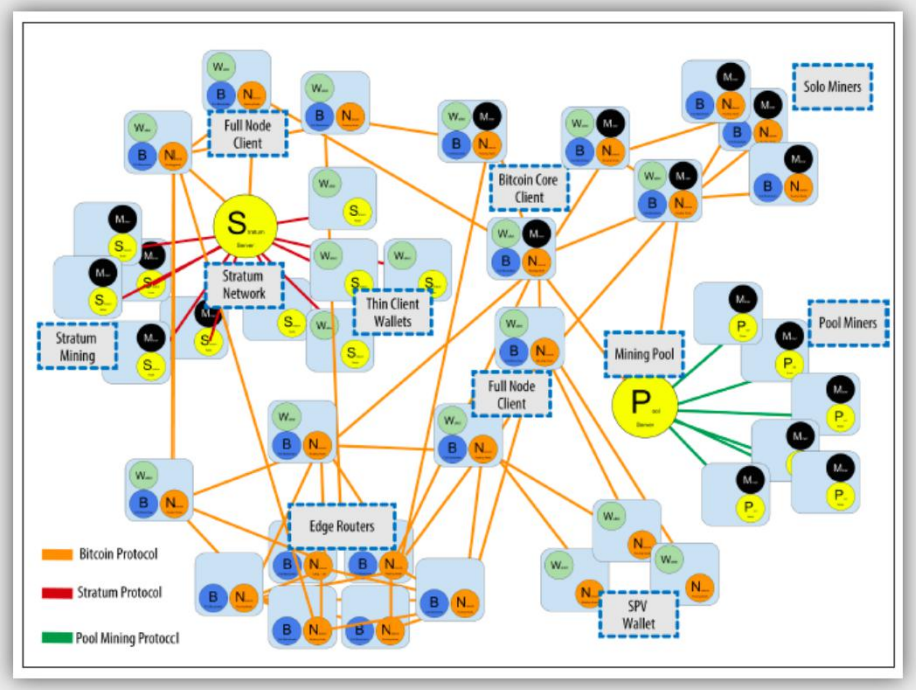
P2P网络 —— 去中心化的网络系统

P2P网络。英文是peer to peer，所以也叫对等网络。顾名思义也就是网络中的每台计算机是对等的，各个节点共同提供网络服务。

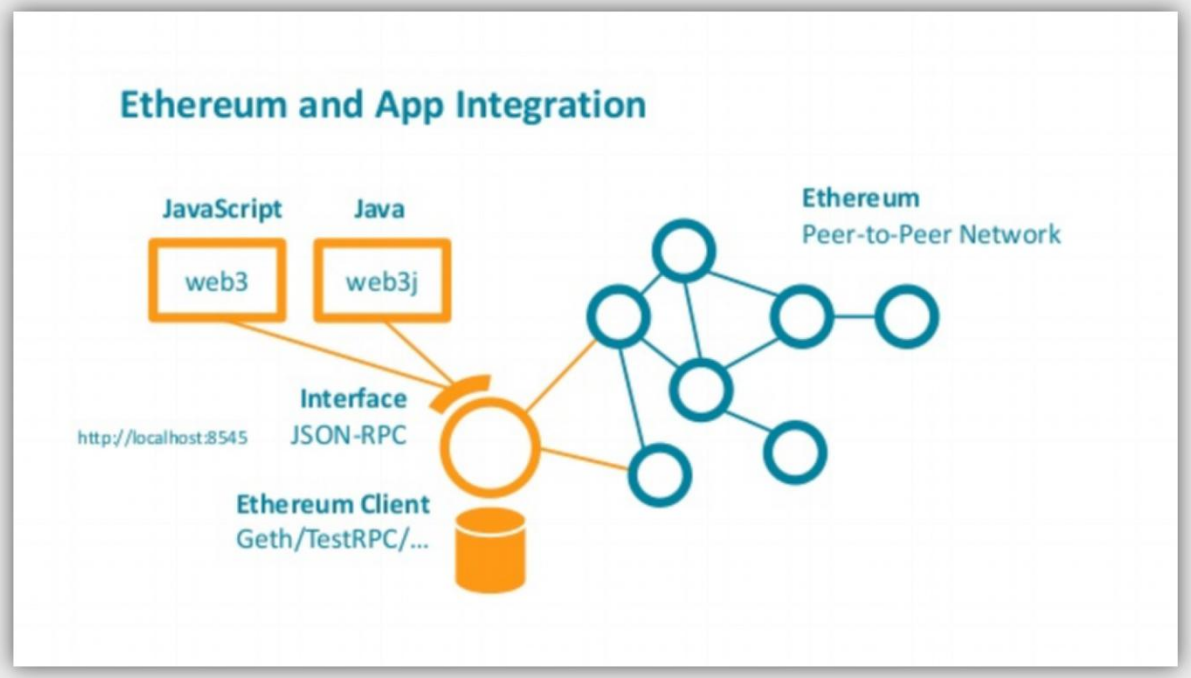
在 P2P 网络中不存在任何中心化的服务器、中心化的服务。这样网络中的任何一台机器出现问题都不会影响其他节点，保证P2P网络的正常运行。



区块链是以P2P技术为基础的永不停止的全球网络

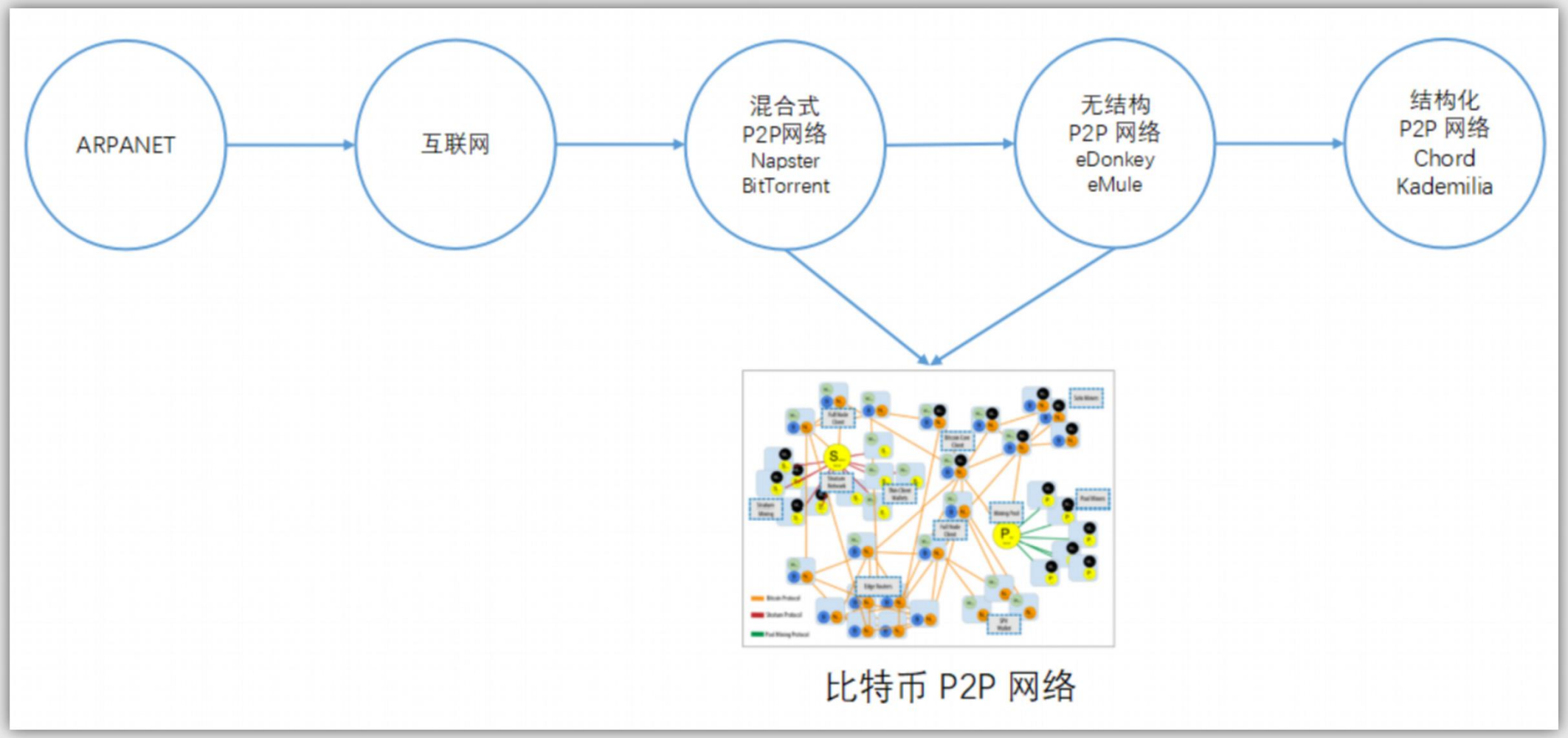


比特币网络

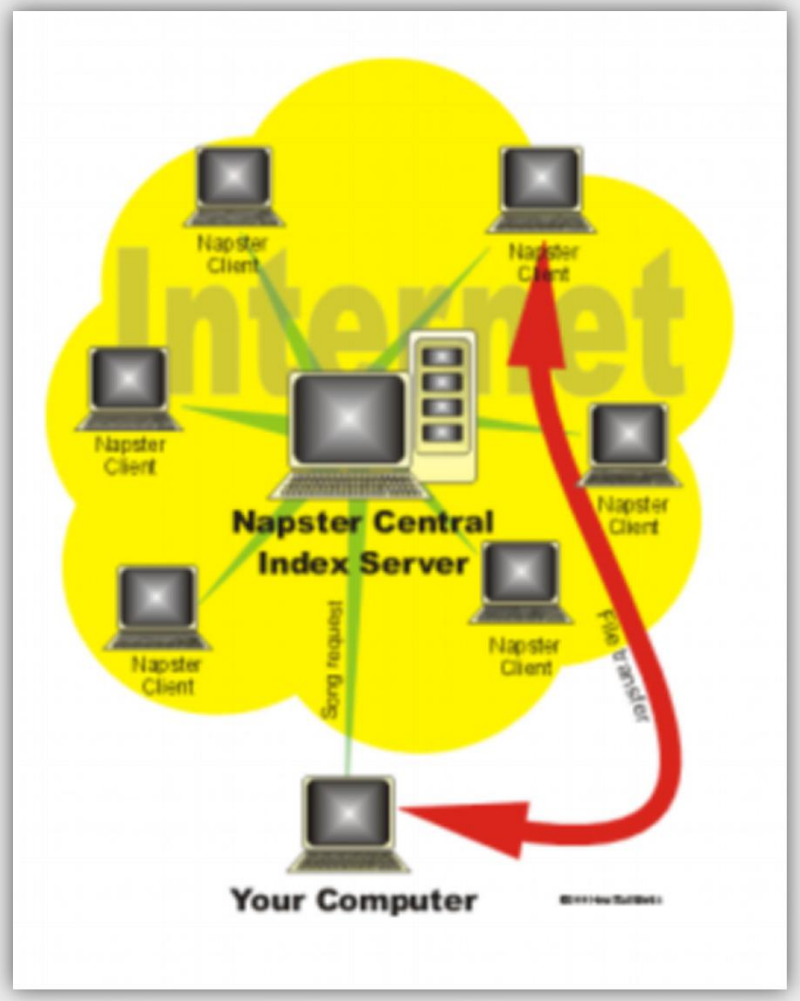


以太坊网络

P2P 网络技术发展脉络

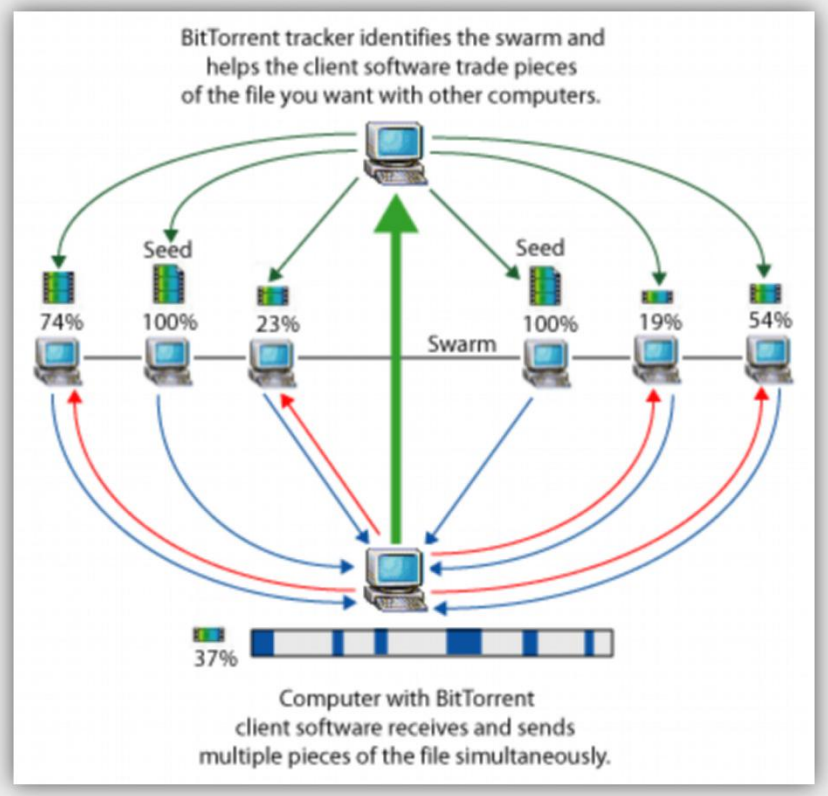


混合式P2P网络 —— Nasper



1. 由中心化的索引服务器(index server)和对等的用户计算机节点(client) 组成，故称为混合式 P2P 网络
2. 每一个新的 client 节点加入，即向 index server 上传自己拥有的 mp3 文件列表
3. 当 client A 试图下载某一个文件 x.mp3 时，向 index server 查询 x.mp3，index server 查到 client B 有 x.mp3，则向 client A 告知 client B 的 IP 地址
4. Client A 与 client B 直接建立连接，并下载 x.mp3
5. 下载完毕后，index server 记录 client A 也拥有 x.mp3
6. Index server 只存储索引，不存储音乐文件，以此规避法律风险，但稍后仍被唱片公司告侵权

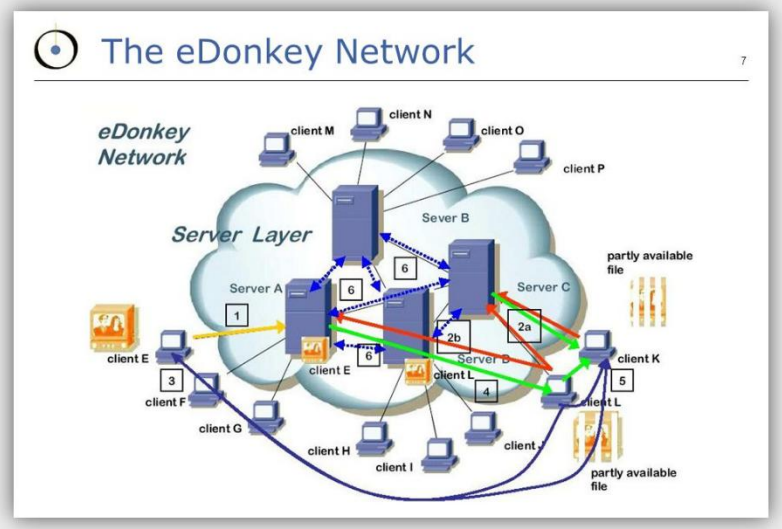
混合式P2P网络 — BitTorrent



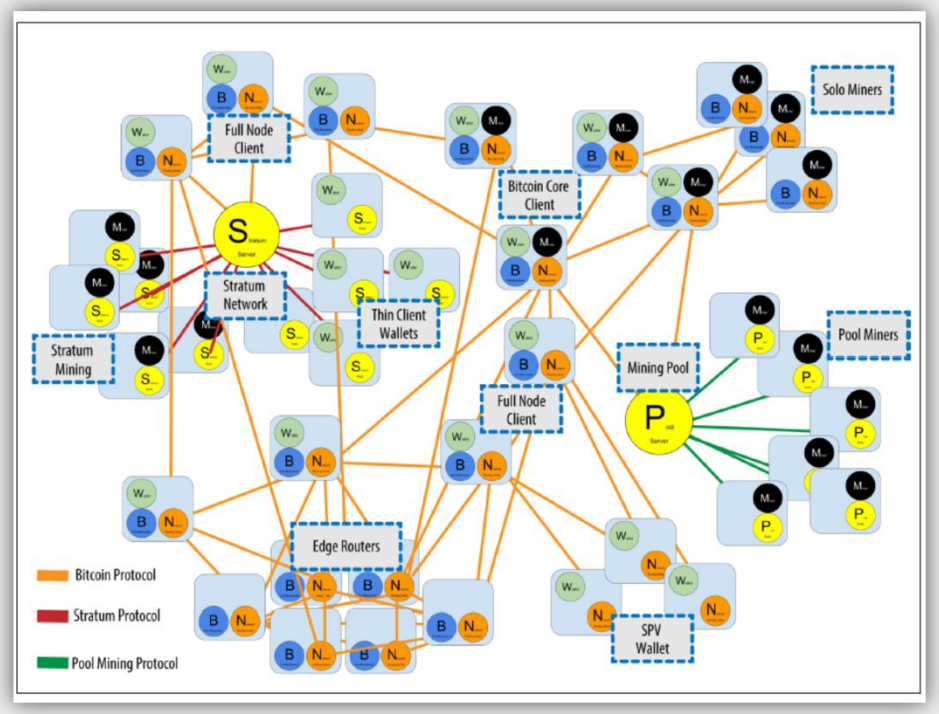
1. BT client 下载某一个文件 x.mp4 的种子文件 x.torrent，x.torrent 中包含该文件 tracker 服务器地址
2. BT client 连接该 tracker 服务器，并被 tracker 编入关于 x.mp4 文件的 swarm 子网当中，拥有同一个文件的节点组成swartm子网
3. 一个 BT client 立刻与 swarm 中多个对等 BT clients 建立连接，分片并行下载文件
4. 某一片文件下载完毕之后，立刻可向同一个 swarm 中的其他节点提供下载
5. 同一个 swarm 中相互连接的节点越多，下载速度越快

无结构P2P网络

1. 综合改进了 Napster 和 BitTorrent
2. 由多个 Index Server 构成 Server 集合，提供索引服务
3. Client 节点之间经 Index Server 引导建立直接连接
4. 文件分片、分段，多片并行下载
5. Server 和 Client 节点自由进出



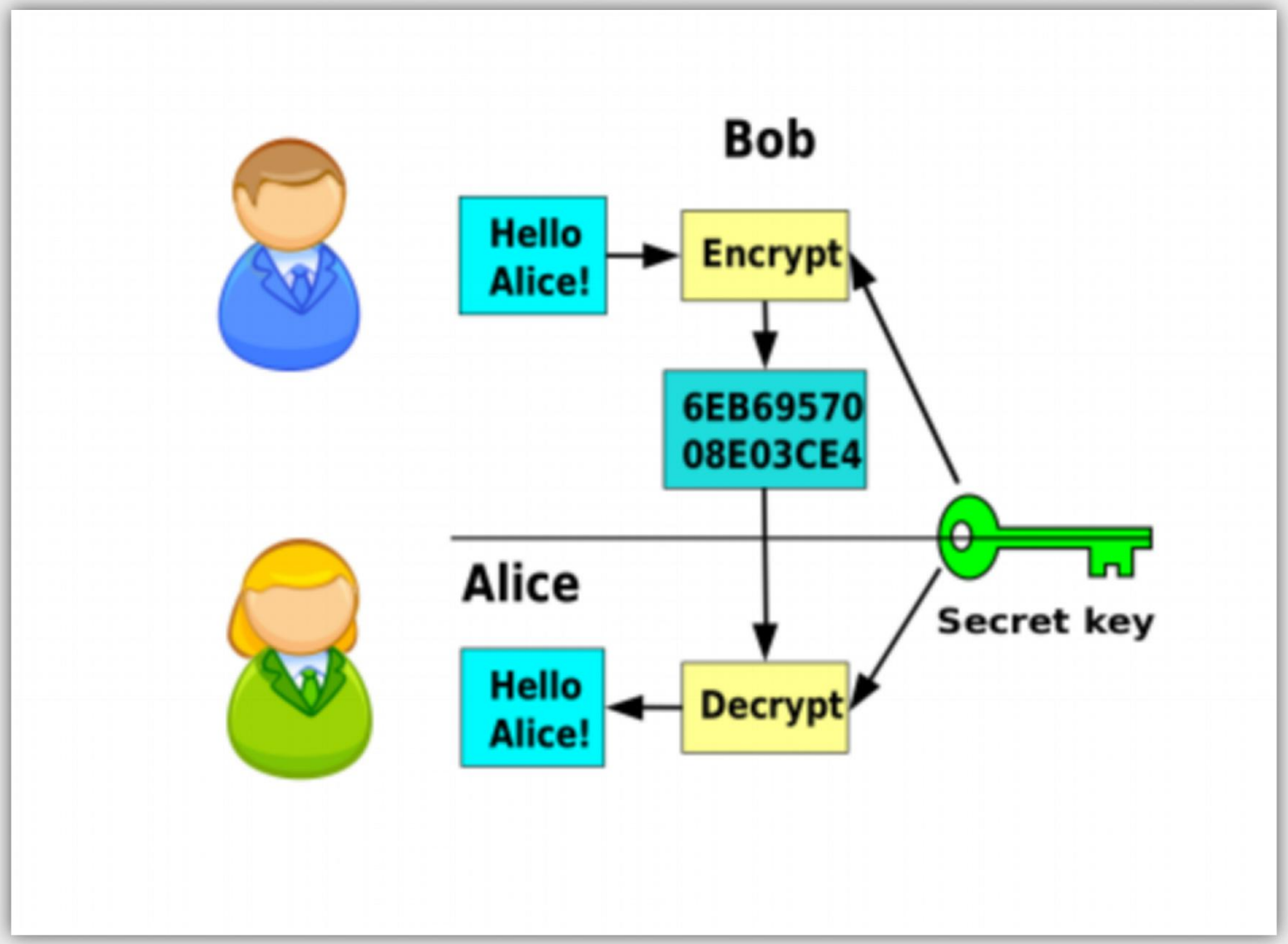
比特币 P2P 网络



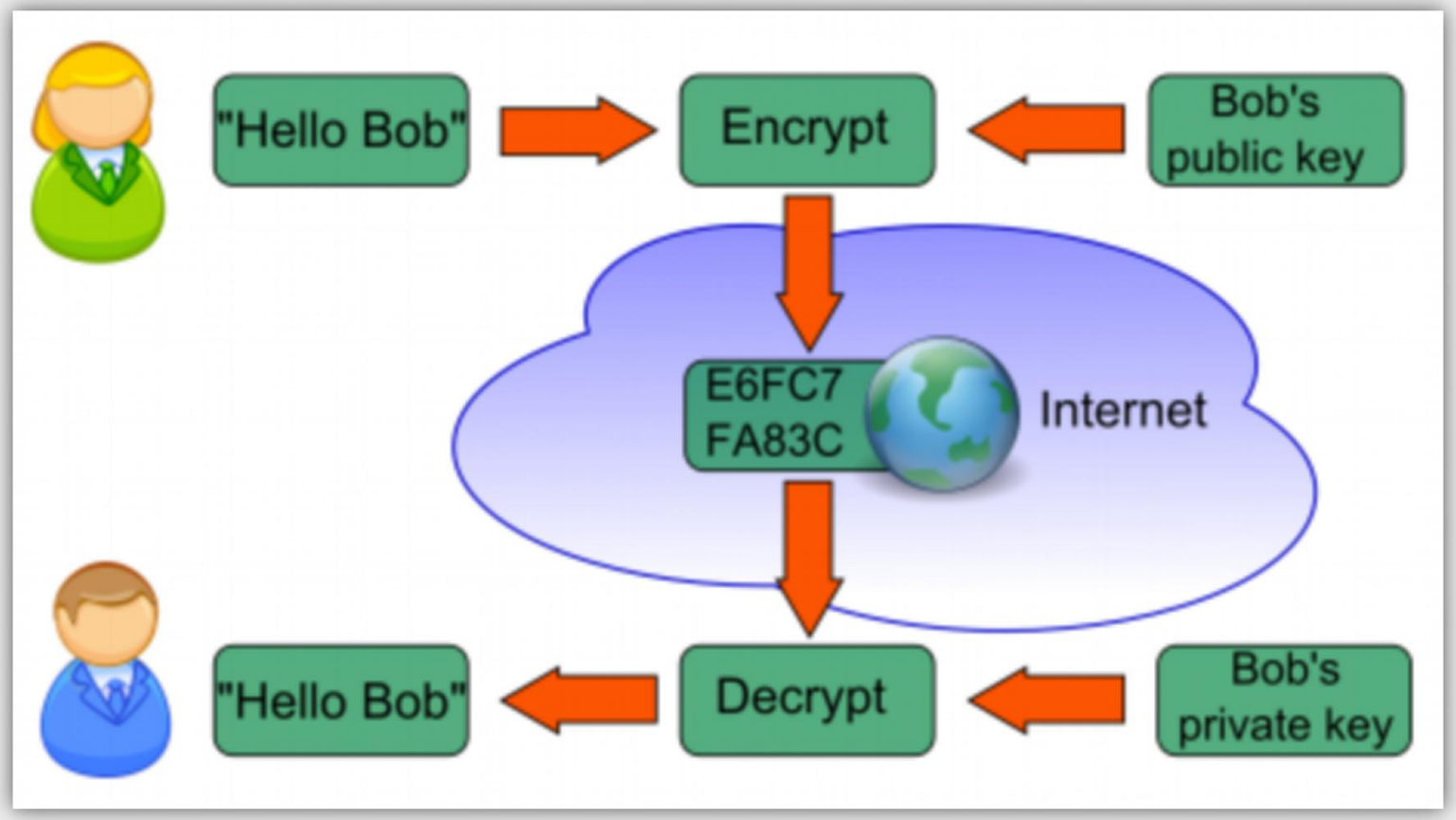
比特币 P2P 网络

1. 新节点通过内置的 DNS 种子节点查询网络 IP 列表
2. 某些种子节点返回一组比特币节点 IP
3. 新节点选择 8 个节点连接，并彼此对比，同步区块链数据
4. 新交易发生时，收到交易的节点向自己所有的邻居节点广播交易消息，后者进一步向自己所有的邻居广播，如此下去，直到全网均受到新交易信息（flooding 算法）
5. 由于每个节点都拥有全部区块链数据，因此无需复杂的路由算法

对称加密通讯

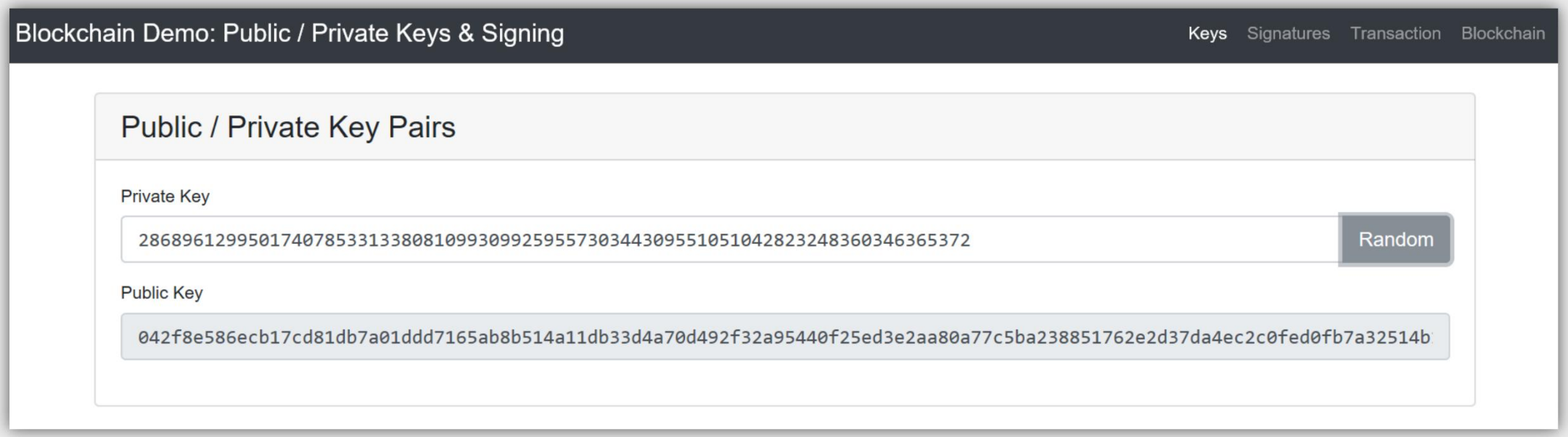


非对称加密和公钥密码学



Demo: 公私钥对

演示地址： https://anders.com/blockchain/public-private-keys/keys.html

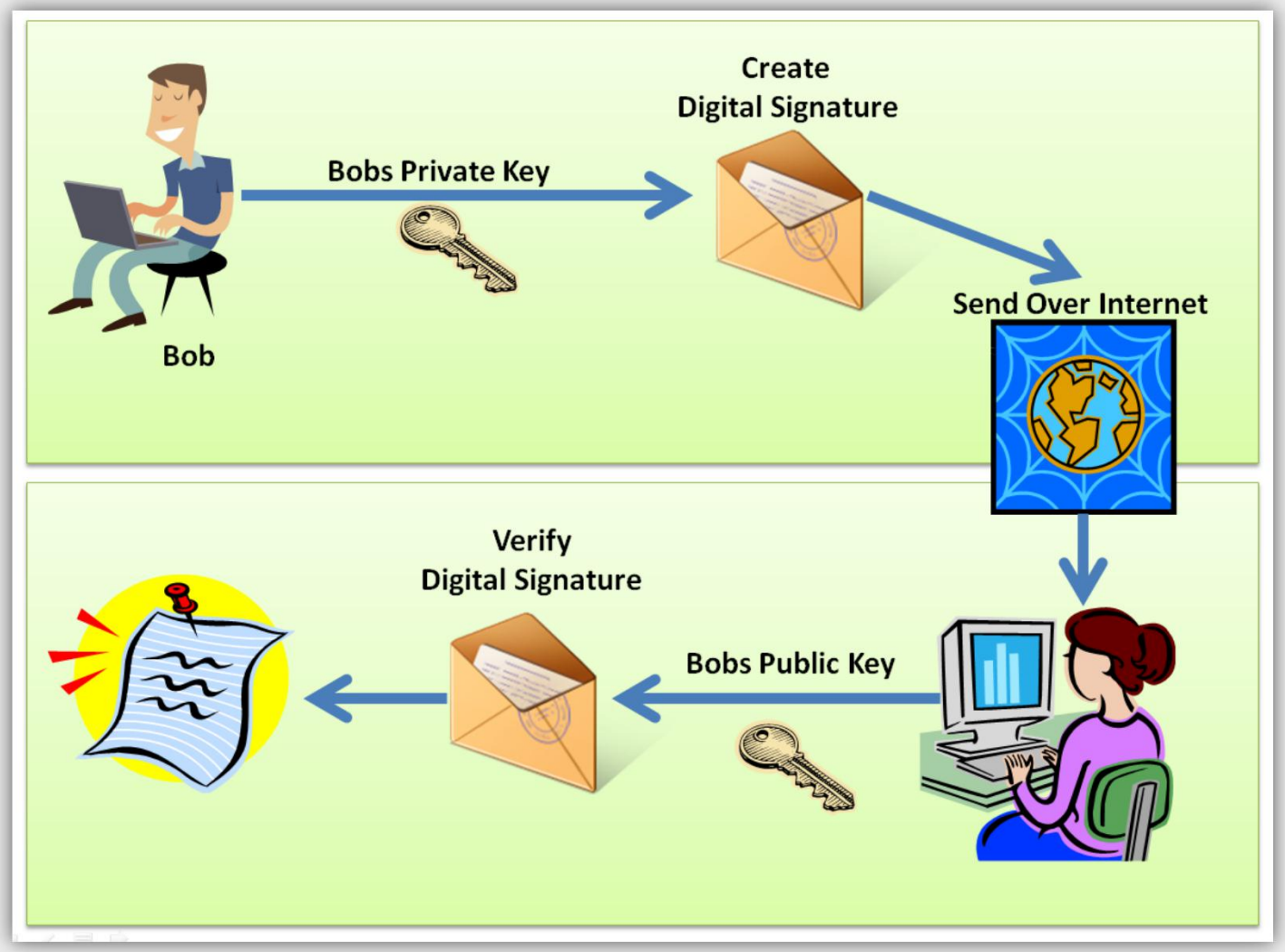


数字签名技术

签名技术的基本要求：

1. 只有你能签名，但任何其他人都可以验证
2. 签名只对某一个特定文件有效，不能复制到其他文件中

基于公钥密码学的数字签名解决方案



数字签名的 API

1. (sk, pk) := generateKeys(keysize)

   sk: 签名私钥

   pk: 验证公钥

2. sig := sign(sk, message)

3. isValid := verify(pk, message, sig)

数字签名方案的要求

1. 合法签名能正确验证

   verify(pk, message, sign(sk, message)) == true

2. 无法伪造签名

   即使攻击者知道公钥（pk），并且可以要求签名者对任意数量的信息进行签名，然后加以各种研究，也无法对一条新的信息伪造出一个合格签名

公钥作为地址，实现去中心化身份认证

1. 公钥 pk == 身份 ID

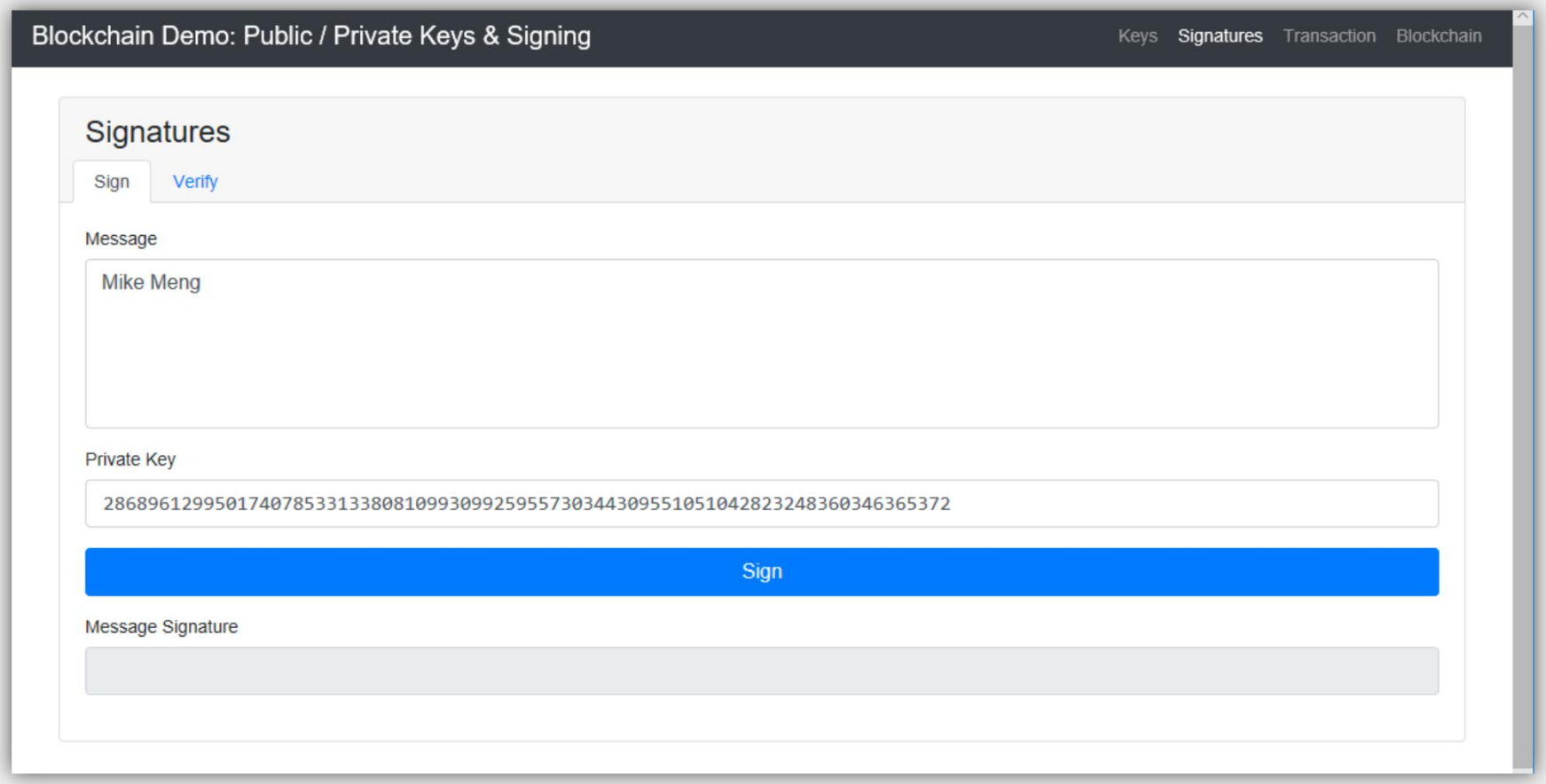
2. 如果你看到一个数字签名 *sig* 使得 *verify(pk, msg, sig) == true*，
3. 则等同于对应于私钥 *sk* 的配对公钥 *pk* 说 *[msg]*

4. 反之，若要以 pk 身份发言，就必须知道对应的 sk

5. *所以，可以将公钥作为匿名网络中的去中心化身份认证*

Demo: 数字签名

演示地址：https://anders.com/blockchain/public-private-keys/signatures.html



总结

1. 比特币的底层通信是P2P网络
2. 数字签名利用了非对称加密技术

布置作业

- 必做内容：
- 理解数字签名原理

EDU

CSDN学院　IT实战派