第一部分 概览

一、计算机与网络安全概念

1、密码算法与协议分为四个主要领域：

对称加密：加密任意大小的数据块或数据流的内容，包括消息、文件、加密密钥和口令

非对称加密：加密小数据库，如加密密钥或数字签名中使用的散列函数值；

数据完整性算法：保护数据块的内容不被修改；

认证协议：基于密码算法设计的认证方案，用来认证实体的身份。

2、计算机安全三个关键目标

保密性、完整性、可用性

3、

二、数论基础

1、整除性和带余除法

B除a用b|a来表示，即a处以b没有余数。

2、欧几里得算法

最大公因数gcd(a,b) = gcd(b,a mod b)

3、模运算

如果a是一个整数，n是正整数，则定义a除以n的余数为a模n，整数n位模数。

若(a mod n) = (b mod n)，则称整数a和b是模n同余的，表a = b(mod n)

4、费马定理和欧拉定理

费马定理和欧拉定理在公钥密码学中占有重要地位。

费马定理，若p是素数，a是正整数且不能被p整除，则

第二部分 对称密码

三、传统加密技术

对称密码模型，也称传统加密或单钥加密

对称密码模型有五个基本成分：明文、加密算法，密钥，密文，解密算法。

明文输入->加密算法->解密算法->明文输出

四、分组密码和数据加密标准

1、流密码与分组密码

流密码每次加密数据流的一位或一个字节。分组密码是将一个明文分组作为作为整体加密并且通常得到的是与明文等长的密文分组。典型分组大小是64位或128位。

2、DES加密

对于任意加密方案，总有两个输入：明文和密钥。DES明文长为64位，密钥长为56位（另有8为作为奇偶校验）。

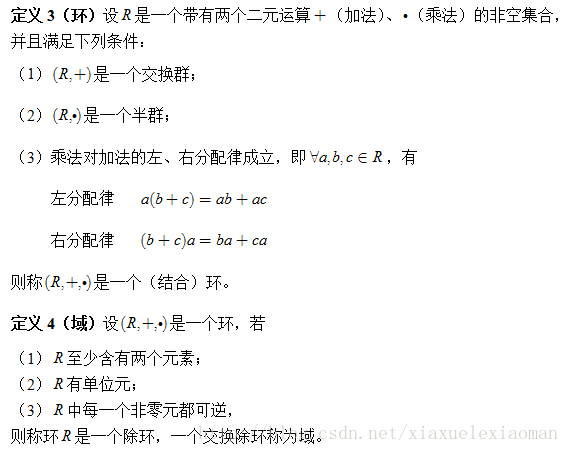
明文或密钥的微小改变将对密文较大的影响是一个好的性质，称之为雪崩效应。

五、有限域

1、群、环、域

群G,有时记作{G,\*}，是定义了一个二元运算的集合,这个二元运算可表示为.，G中的每一个序偶(a,b)通过运算生成G中的元素(a.b)。

环R，有时记作{R,+,\*}是一个有两个二元运算的集合，这两个运算分布是加法和乘法。



域F，有时记作{F,+,\*}是一个有两个二元运算的集合，这两个运算分布是加法和乘法。

在信息安全中比较有用的是有限域，主要有素域、二进制域等等。

在抽象代数中，[域](https://baike.baidu.com/item/%E5%9F%9F/12740242)是一种可进行加、减、乘和除运算的代数结构。域的概念是[数域](https://baike.baidu.com/item/%E6%95%B0%E5%9F%9F)以及[四则运算](https://baike.baidu.com/item/%E5%9B%9B%E5%88%99%E8%BF%90%E7%AE%97)的推广。域是[环](https://baike.baidu.com/item/%E7%8E%AF/12795512)的一种。域和一般环的区别在于域要求它的元素可以进行除法运算，这等价于每个非零的元素都要有乘法逆元。

实际上，[域](https://baike.baidu.com/item/%E5%9F%9F/12740242)是一个可以在其上进行加法、减法、乘法和除法运算而结果不会超出域的集合。如[有理数](https://baike.baidu.com/item/%E6%9C%89%E7%90%86%E6%95%B0)集合、[实数](https://baike.baidu.com/item/%E5%AE%9E%E6%95%B0)集合、[复数](https://baike.baidu.com/item/%E5%A4%8D%E6%95%B0/254365)集合都是域，但整数集合不是（很明显，使用除法得到的分数或小数已超出整数集合）。

六、高级加密标准

AES是一个对称分组密码算法，2001发布的高级加密标准。

严格地说，AES和Rijndael加密法并不完全一样（虽然在实际应用中二者可以互换），因为Rijndael加密法可以支持更大范围的[区块](https://baike.baidu.com/item/%E5%8C%BA%E5%9D%97)和[密钥长度](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5%E9%95%BF%E5%BA%A6)：AES的区块长度固定为128 [比特](https://baike.baidu.com/item/%E6%AF%94%E7%89%B9)，密钥长度则可以是128，192或256比特；而Rijndael使用的密钥和区块长度可以是32位的整数倍，以128位为下限，256比特为上限。加密过程中使用的密钥是由[Rijndael密钥生成方案](https://baike.baidu.com/item/Rijndael%E5%AF%86%E9%92%A5%E7%94%9F%E6%88%90%E6%96%B9%E6%A1%88)产生。

大多数AES计算是在一个特别的[有限域](https://baike.baidu.com/item/%E6%9C%89%E9%99%90%E5%9F%9F)完成的。

AES加密过程是在一个4×4的[字节](https://baike.baidu.com/item/%E5%AD%97%E8%8A%82)矩阵上运作，这个矩阵又称为“状态（state）”，其初值就是一个明文区块（矩阵中一个元素大小就是明文区块中的一个Byte）。（Rijndael加密法因支持更大的区块，其矩阵行数可视情况增加）加密时，各轮AES加密循环（除最后一轮外）均包含4个步骤：

AddRoundKey — 矩阵中的每一个字节都与该次轮秘钥（round key）做XOR运算；每个子密钥由密钥生成方案产生。

**DES与AES的比较**

自DES 算法公诸于世以来，学术界围绕它的安全性等方面进行了研究并展开了激烈的争论。在技术上，对DES的批评主要集中在以下几个方面：

1、作为分组密码，**DES 的加密单位仅有64 位二进制**，这对于数据传输来说太小，因为每个分组仅含8 个字符，而且其中某些位还要用于奇偶校验或其他通讯开销。

2、DES 的密钥的位数太短，只有56 比特，而且各次迭代中使用的密钥是递推产生的，这种相关必然降低密码体制的安全性，在现有技术下用穷举法寻找密钥已趋于可行。

3、**DES 不能对抗差分和线性密码分析**。

4、DES 用户实际使用的密钥长度为56bit，理论上最大加密强度为256。DES 算法要提高加密强度（例如增加密钥长度），则系统开销呈指数增长。除采用提高硬件功能和增加并行处理功能外，从算法本身和软件技术方面都无法提高DES 算法的加密强度。

相对DES算法来说，AES算法无疑解决了上述问题，主要表现在以下几个方面：

1、**运算速度快**，在有反馈模式、无反馈模式的软硬件中，Rijndael都表现出非常好的性能。

2、对内存的需求非常低,适合于受限环境。

3、Rijndael 是一个分组迭代密码， 分组长度和密钥长度设计灵活。

4、AES标准支持**可变分组长度**，分组长度可设定为32 比特的任意倍数，最小值为128 比特，最大值为256 比特。

5、AES的密钥长度比DES大， 它也可设定为32 比特的任意倍数，最小值为

6、**12 比特，最大值为256 比特，所以用穷举法是不可能破解的。**

7、AES算法的**设计策略是WTS。WTS 是针对差分分析和线性分析**提出的，可对抗差分密码分析和线性密码分析。

**DES与RSA的比较**

RSA算法的密钥很长，具有较好的安全性，但加密的计算量很大，加密速度较慢限制了其应用范围。为减少计算量，在传送信息时，常采用传统加密方法与公开密钥加密方法相结合的方式，即信息采用改进的DES对话密钥加密，然后使用RSA密钥加密对话密钥和信息摘要。对方收到信息后，用不同的密钥解密并可核对信息摘要。

**采用DES与RSA相结合的应用，使它们的优缺点正好互补**，即DES加密速度快，适合加密较长的报文，可用其加密明文；RSA加密速度慢，安全性好，应**用于DES 密钥的加密**，可解决DES 密钥分配的问题。

目前这种RSA和DES结合的方法已成为EMAIL保密通信标准。

七、分组加密的工作模式

多重加密与三重DES

八、伪随机数的产生

第三部分 公钥密码

九、公钥密码学和RSA

公钥密码学与传统密码学的区别

1、公钥算法是基于数学函数；传统密码学算法是基于代替和置换

2、公钥密码是非对称的，它使用两个独立的密钥；传统密码学使用一个密钥

3、任何加密方法的安全性依赖于密钥的长度和破译密钥所需要的计算量

4、公钥密码学仅限于应用在密钥管理和数字签名中

5、使用公钥密码也需要某种形式的协议，该协议通常包含 一个中心代理，并且其所包含的处理过程既不比传统密码的那些过程简单，也不比之更有效。

公钥密码术语

1、非对称密钥

两个密钥：公钥和私钥，用来实现互补运算，即加密和解密，或者生成签名和验证签名

2、公钥证书

认证机构将用户的姓名和公钥绑定在一起，用户用自己的私钥对数字文件签名后，可以通过证书识别签名者，因为签名者是唯一拥有证书与证书上对应的私钥的用户

3、公钥密码（非对称密码）算法

含有两个密钥：公钥和私钥。从公钥中推出私钥在计算上不可行

4、公钥基础PKI

设施有一系列的协议、服务平台、软件和工作站组成，用于管理证书和公钥-私钥对，并产生、维护和废除公钥证书

公钥密码体制的基本原理

产生原因 ：对称密码体制的密钥分配问题 ，对数字签名的需求

六个组成部分

明文：算法的输入。原始的消息或数据。

加密算法：对明文进行各种转换。

公钥和私钥：算法的输入。这对密钥中一个用于加密，一个用于解密。加密算法执行的变换依赖于公钥或者私钥。

密文：算法的输出，依赖于明文和密钥，对给定的消息，不同密钥产生的密文不同。

解密算法：接收密文和相应的密钥，并产生原始的明文。

加解密步骤

每一用户产生一对密钥，用来加密和解密消息

每一用户将其中一个密钥存于公开的寄存器或者其他可访问的文件中，该密钥称为公钥，令一密钥是私有的

若Bob要发消息给Alice，则BOb用Alice的公钥对消息加密

Alice收到消息后，用其私钥对消息解密。由于只有Alice知道其自身的密钥，所以其他的接受者均不能解密出消息

利用这种方法，通信各方均可访问公钥，而私钥是各通信方在本地产生的，所以不必进行分配。只要用户的私钥收到保护，保持秘密性，那么通信就是安全的。在任何时刻，系统可以改变其私钥，并公布相应的公钥以代替原来的公钥。

两个密钥中，使用加密的密钥不同，实现的功能也不同：

* 接收方公钥用来加密，实现保密性
* 发送方私钥用来加密，实现认证性

发送方首先用**其私钥对消息加密，得到数字签名，然后在用接收方的公钥加密**，所得到的密文只能被拥有相应私钥的接收方解密，这样**既可以保证消息的保密性，也能提供认证功能**。但是这种方法的缺点在于：在每次通信中要执行4次复杂的公钥算法。

**公钥加密体制的应用**

* **加密/解密**：**发送方用接收方的公钥对消息加密**
* **数字签名**：发送方用其**私钥对消息“签名”**
* **密钥交换**：通信双方交换会话密钥

公钥密码体制的优缺点

优点：

1、解决了对称密码体制密钥分发问题

2、发送方和接收方没有必要共享密钥，一个人只需要有一个秘密密钥就能能一群人进行安全的交流。

3、实现数字签名/密钥交换

缺点：

计算的时间长，时间复杂度高，降低了加解密效率

因此，公钥加密系统不用于加密大量数据，而是用于密钥管理和签名应用程序

[公钥，私钥和数字签名这样最好理解](https://www.cnblogs.com/shenkebky/p/8461841.html)

**1、**公钥和私钥是成对的，它们互相解密。

2、一般的公钥加密，私钥解密。

3、私钥数字签名，公钥验证。用私钥加密，对方通过公钥解密即可知道对方是谁（**因为私钥可以确定身份**）。

例如我的好朋友x说有人冒充我给他发信。怎么办呢？我把我要发的信，内容是c，用我的私钥2，加密，加密后的内容是d，发给x，再告诉他 解密看是不是c。他用我的公钥1解密，发现果然是c。 这个时候，他会想到，能够用我的公钥解密的数据，必然是用我的私钥加的密。只有我知道我得私钥，因此他就可以确认确实是我发的东西。 这样我们就能确认发送方身份了。这个过程叫做数字签名。

十、密钥管理和其他公钥密码体制

第四部分 密码学中的数据和完整性算法

十一、密码学Hash函数

哈希函数是一种函数。

举一个简单的例子来说明它的用法。譬如学校里以学生名字为索引建了一张表，存储学生的信息。同时希望在查找的时候能够高效率地查找到学生。可以怎么实现呢？（如果以学号为索引，可以一次查找就找到；以名字为索引纯粹是举例；但是在实际中有很多以名字为索引的情况，譬如Web开发中，表单form的各项一般都有名字，在后台代码实现中便是以名字为索引的）

如果靠匹配字符串来查找，毫无疑问是非常低效的；所以这个时候的一个做法就是，首先在存储时，对名字进行一次哈希，得到一个数字；然后将名字存在相应数字所对应的位置。在进行查找时，再对希望查找的名字进行一次哈希，得到位置。在算法上，哈希的平均复杂度是O(1)。

加密哈希（cryptographic hash）

加密哈希是哈希函数的一种，它具有某些属性，使其适用于加密。通过加密哈希，它将任意大小的数据映射到固定大小的位串（散列），并被设计为单向函数，即一个不可逆的函数。从理想的加密散列函数输出重新创建输入数据的唯一方法是尝试对可能的输入进行暴力搜索，以查看它们是否产生匹配，或使用匹配哈希的彩虹表。布鲁斯施奈尔称单向散列函数是“现代密码学的主力”。输入数据通常称为消息（Message），输出（散列值或散列）通常称为消息摘要或简称为摘要（Digest）。

理想的加密哈希函数有五个主要属性：

1.它是确定性的，因此相同的消息总是产生相同的散列

2. 可以快速计算任何给定消息的哈希值

3. 除了通过尝试所有可能的消息之外，从其散列值生成消息是不可行的

4. 对消息进行小的更改便能引起哈希值的巨大改变，以使新哈希值看起来与旧哈希值不相关

5. 找到具有相同散列值的两个不同消息是不可行的

SHA-256是SHA-2中的一个算法。SHA-2，也即第二代安全散列算法（Secure Hash Algorithm 2），由美国国家安全局2001年公布的标准哈希算法，是SHA-1的后继。

SHA-2下包括六个不同的算法标准：SHA-224、SHA-256、SHA-384、SHA-512、SHA-512/224、SHA-512/256。这些不同的算法使用不同生成摘要的长度 、循环运行的次数，但算法的基本结构是一致的。现在已知SHA-2容易受到长度扩展攻击，所以推荐使用SHA-3来取代它。

对于任意长度的消息，SHA256都产生256bit长的哈希值，也即32字节，或者64位的16进制数，或者8个8位的16进制数 。

哈希在区块链中的用处包括：

利用puzzle-friendliness的工作量证明，也即，要求生成一个新区块时，哈希值一定要是以一定数量的连续的0开始的；

对每个区块的内容进行哈希，后面的区块相当于一直在对前面的区块进行哈希，所以一般来说，如果一个区块之后如果跟上其他6个区块之后，一般认为该区块的内容已经被公认了，不可更改。

第五部分 区块链

1、基本概念

区块链是分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。所谓共识机制是区块链系统中实现不同节点之间建立信任、获取权益的数学算法。

区块链（Blockchain）是比特币的一个重要概念，它本质上是一个去中心化的数据库，同时作为比特币的底层技术。区块链是一串使用密码学方法相关联产生的数据块，每一个数据块中包含了一次比特币网络交易的信息，用于验证其信息的有效性（防伪）和生成下一个区块。

狭义来讲，区块链是一种按照时间顺序将数据区块以顺序相连的方式组合成的一种链式数据结构， 并以密码学方式保证的不可篡改和不可伪造的分布式账本。

广义来讲，区块链技术是利用块链式数据结构来验证与存储数据、利用分布式节点共识算法来生成和更新数据、利用密码学的方式保证数据传输和访问的安全、利用由自动化脚本代码组成的智能合约来编程和操作数据的一种全新的分布式基础架构与计算方式。

区块链主要解决的交易的信任和安全问题，因此它针对这个问题提出了四个技术创新：

（1）分布式账本，就是交易记账由分布在不同地方的多个节点共同完成，而且每一个节点都记录的是完整的账目，因此它们都可以参与监督交易合法性，同时也可以共同为其作证。

跟传统的分布式存储有所不同，区块链的分布式存储的独特性主要体现在两个方面：一是区块链每个节点都按照块链式结构存储完整的数据，传统分布式存储一般是将数据按照一定的规则分成多份进行存储。二是区块链每个节点存储都是独立的、地位等同的，依靠共识机制保证存储的一致性，而传统分布式存储一般是通过中心节点往其他备份节点同步数据。没有任何一个节点可以单独记录账本数据，从而避免了单一记账人被控制或者被贿赂而记假账的可能性。也由于记账节点足够多，理论上讲除非所有的节点被破坏，否则账目就不会丢失，从而保证了账目数据的安全性。

（2）非对称加密和授权技术，存储在区块链上的交易信息是公开的，但是账户身份信息是高度加密的，只有在数据拥有者授权的情况下才能访问到，从而保证了数据的安全和个人的隐私。

（3）共识机制，就是所有记账节点之间怎么达成共识，去认定一个记录的有效性，这既是认定的手段，也是防止篡改的手段。区块链提出了四种不同的共识机制，适用于不同的应用场景，在效率和安全性之间取得平衡。区块链的共识机制具备“少数服从多数”以及“人人平等”的特点，其中“少数服从多数”并不完全指节点个数，也可以是计算能力、股权数或者其他的计算机可以比较的特征量。“人人平等”是当节点满足条件时，所有节点都有权优先提出共识结果、直接被其他节点认同后并最后有可能成为最终共识结果。以比特币为例，采用的是工作量证明，只有在控制了全网超过51%的记账节点的情况下，才有可能伪造出一条不存在的记录。当加入区块链的节点足够多的时候，这基本上不可能，从而杜绝了造假的可能.

（4）智能合约，智能合约是基于这些可信的不可篡改的数据，可以自动化的执行一些预先定义好的规则和条款。以保险为例，如果说每个人的信息（包括医疗信息和风险发生的信息）都是真实可信的，那就很容易的在一些标准化的保险产品中，去进行自动化的理赔.

比特币在设计之初就希望在不动用警察军队等暴力机关的前提下让攻击者自己觉得攻击毫无意义。事实上它确实做到了

* 偷取比特币   
  别人转给他的钱必须有私钥方能解锁花费。
* 伪造比特币   
  任何比特币不能无中生有，来源就两个！伪造者将无法解锁

去中心化、高效、安全，不可能实现三者全部同时达到极致。

2、区块链和密码学

在比特币的世界，没有属于个人的账户的概念，只有比特币钱包，所谓的比特币钱包，其实可以简单理解为一个公钥。由于电子计算机内的任何东西都是可以零成本复制的，所以就必须采用非对称密码技术解决这个问题。

  简单点说，一个钱包的含义就是“任何人都可以放钱进去，但只有你自己可以拿钱出来”，你之所以比别人多一个拿钱出来花的权力，是因为你持有和钱包公钥对应的私钥！请务必记住，私钥就是你打开钱包花钱的钥匙，一旦被别人窃取，钱也就是别人的了，一旦丢失或者销毁，钱包里的钱也就相当于跟着销毁了，就好像现实中你把纸钞撕毁焚烧了一样。

既然没有账户的概念，我怎么知道我还有多少比特币可以花费呢？

  在比特币世界，所有能花费的钱被统一表示成UTXO(即Unspent Transaction Output)，该UTXO和一个钱包地址关联，你要想知道你还有多少钱可以花，你只能遍历你所拥有的钱包，然后把和该钱包关联的UTXO累加在一起，结果便是你的“账户余额。所有的这一切信息，全部分布式存储在已经达成全网共识的区块链里

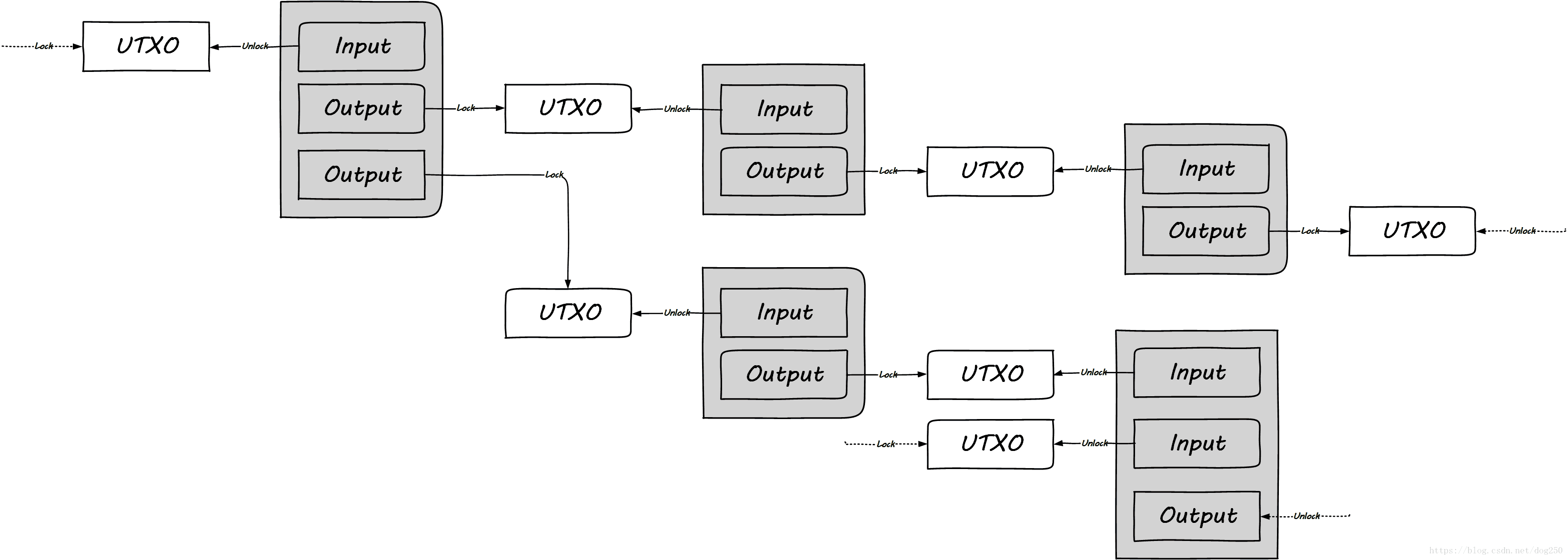
UTXO是比特币世界基本的消费单位，就像我们日常的人民币面值一样，你无法将一张10块钱撕开成两半花，你若买一个5块钱的东西，你必须付出整个10块钱，然后接受找零的5块钱，当然，你也可以说“不用找了！

**非对称密码技术**的两个要点：

* 公钥加密，私钥解密   
  **确保信息的接收者**只能是确定的人。
* 私钥签名，公钥验签   
  **确保信息的发送者**只能是确定的人。

**区块链的安全性完全依赖于数字签名技术的安全性，即ECC算法和RSA算法等基于三类数学问题的难解性**：大整数分解问题、离散对数问题以及椭圆曲线问题。限于当今计算机能力的限制，目前还没有能完全攻克他们的算法，所以**当前来说**是较为安全的。

花钱的时候，必须先解锁这笔封存在区块链里的钱，再上锁发往它的目的地。依此循环不止，整个世界的比特币交易从中本聪创世开始，输入/输出的解锁/锁定关系就形成了下面的有向无环图：



3、区块链和哈希密码

哈希算法是区块链中用的最多的一种算法，它被广泛的使用在构建区块和确认交易的完整性上。

它是一类数学函数算法，又被称为散列算法，需具备三个基本特性：

其输入可为任意大小的字符串

它产生固定大小的输出

它能进行有效计算，也就是能在合理的时间内就能算出输出值

如果要求哈希算法达到密码学安全的话，我们还要求它具备以下三个附加特性：

1. 碰撞阻力： 是指对于两个不同的输入，必须产生两个不同的输出。如果对于两个不同的输入产生了相同的输出，那么就说明不具备碰撞阻力，或是弱碰撞阻力。
2. 隐秘性：   
   也被称为不可逆性，是指 y = HASH（x）中，通过输入值x，可以计算出输出值y，但是无法通过y值去反推计算出x值。为了保证不可逆，就得让x的取值来自一个非常广泛的集合，使之很难**通过计算反推出x值**。
3. 谜题友好：   
   这个特性可以理解为，谜题是公平友好的，例如算法中 y = HASH（x），如果已知y值，想去得到x值，**那就必须暴力枚举，不断的尝试才能做到，并且没有比这更好的办法，没有捷径。**

我们以比特币为例，来看一下哈希算法的具体应用：   
在比特币中，使用**哈希算法把交易生成数据摘要，当前区块里面包含上一个区块的哈希值，后面一个区块又包含当前区块的哈希值**，就这样一个接一个的连接起来，形成一个哈希指针链表。

哈希算法可以**检验信息是否是相同的**，这样的优势可以节省重复文件传送的时间。

除此之外，哈希算法还可以检验信息的拥有者是否真实。

**Hash算法特别的地方在于它是一种单向算法，用户可以通过hash算法对目标信息生成一段特定长度的唯一hash值，却不能通过这个hash值重新获得目标信息。因此Hash算法常用在不可还原的密码存储、信息完整性校验**等。

**对称加密和非对称加密**

1. 对称加密

对称加密指的就是加密和解密使用同一个秘钥，所以叫做对称加密。对称加密只有一个秘钥，作为私钥。   
常见的对称加密算法：DES，AES，3DES等等。

2. 非对称加密

非对称加密指的是：加密和解密使用不同的秘钥，一把作为公开的公钥，另一把作为私钥。公钥加密的信息，只有私钥才能解密。私钥加密的信息，只有公钥才能解密。   
常见的非对称加密算法：RSA，ECC

3. 区别

对称加密算法相比非对称加密算法来说，加解密的效率要高得多。但是缺陷在于对于秘钥的管理上，以及在非安全信道中通讯时，密钥交换的安全性不能保障。所以在实际的网络环境中，会将两者混合使用.