Devon Rudd

Muntuck Yap

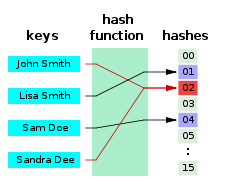
Operating Systems

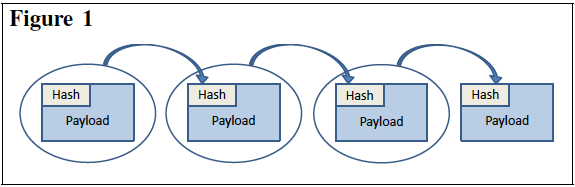
October 15, 2018

Test 3

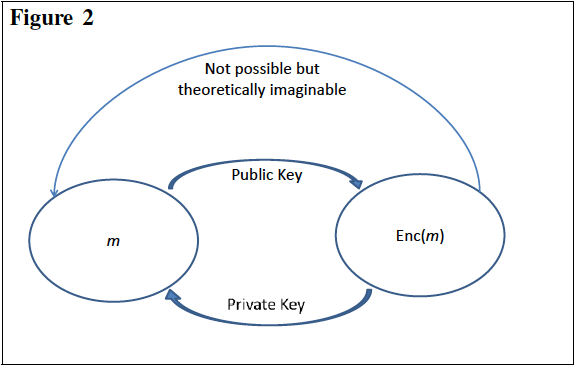
Block Chain Implementation

Block chains are implemented by using a hash chain (aka block) and linking it into another hash chain. Hash chains are a series of hashes that are linked to on another and a hash is the encrypted form of data that is created from a hash function. The way a hash is created is by sending a string (plaintext characters/words) of arbitrary size through a hash function, typically a one way mathematical function (meaning the output can’t be traced back without first knowing the input), and the output returns a number of a fixed size. An example of this is the SHA1 hash which will return a number that is 160 bits in size no matter the length of the string. The picture below from Wikipedia gives an example of a string being converted into a number by way of hashing and also shows an example of a hash collision, where two separate strings can have the same hash value.



Hash chains are homogenous pieces of data (blocks) that are linked together by a hash function. Each block of data contains the hash and the payload (arbitrary size of data). The hash is passed to the next link in the chain and hashed again from the last hash and passed again until the chain ends. In order to decode the entirety of the data you will need to run the hashes in the correct order. Also, if the first payload of the hash chain is changed, then the second one will be changed, and the third and so on. The figure below from Oleg Manzoka’s paper gives a good diagram of this (Blockchain: Simple Explaination). 

The next step towards a block chain is to allow only one person or machine to create a new block in the chain. A solution to this problem is to use public key cryptography (PKC). PKC is similar to hashing as they use one-way functions to encryption, but the difference between the two is that with PKC, a key (a specified string used similar to a password) so that only those with the key can access the data. Private Key Cryptography uses large prime numbers multiplied to compute a private key. It is not impossible to crack the key, but extremely difficult because they are very large. The figure below describes this in mathematical terms of *m* (key) and the encryption function, *Enc(m)*.



Now if we implement PKC with Hash chains inside of hash chains we get a block chain. With this you distribute the data amongst many different machines. These machines have to follow the rules and those that have access that don’t are ignored in order to maintain the integrity of the data. The rules are:

1. Creating block chains require a lot of computational resources.
2. Creating a new block in the chain is rewarding, so many machines would make the effort to create the new block.
3. When a branch happens, the person with the longest branch wins.

This maintains the integrity the data and ensures that it if an error occurs, the original data can be restored.