Blockchain Cryptography

Blockchain is peer-to-peer network every node is link with others



(Peer-to-peer Network)

Napster

Torrent

4 Things which are to be concern when we talk about network.

1:**Confidentiality**

2:**Integrity**

3:**Non Repudiation**

4:**Authentication**

Confidentiality

*Lets we want System A to send some secret message to System B and we don’t want system C can see the message .*

B

**A**



C

Integrity

In integrity we don’t wanted the message we sent to B is tempered by C and then sent to B.



C



Non-Repudiation

**If system A has sent the message to system B then there must has a proof of it that A has sent message.**





Authentication

It should be authenticated that if A sent message to B then B must receive this message only from A not C who can change the name of like A’ message

Here we can solve these 4 problems by using Cryptography

Crypt Secret/Hidden

graphy Writing

The process of cryptography is to convert your message into cipher(encrypted form) which cant read by others but only reciever.

Types of Cryptography

**Cryptography**

Asymmetric Key

Symmetric Key

**Symmetric Key Cryptography**:

In Symmetric key cryptography every **Node(system)** has its own keys like**,k1,k2,k3** and etc.

If a **Node A** wants to send a message to **Node B** than **Node A** encrypt(means that message cant be read by others) the message using key **K1** and Node B also use Key **K1** to decrypt(to read it) it.

No if **Node A** want to send message to **Node C** than it must be used different key not the key which already used by it when interact with **Node B**.

So **Node A** will use key **K2 for** to send message to **Node C**

So there will be bunch of keys which is very difficult to manage and remember

**That’s is the main drawback of Symmetric Key CryptoGraphy**

We see it in detail in following diagrams.

Note: **Cipher text is the converted encrypted form which cant be directly readed.**



**K1**

**Encryption**



**Cipher Text**

**Decryption Using key1**

**Key1**

**Cipher Text**

**Key1**



(Picture1)



**Encryption**

**K1**

Key on Apple iOS 13.3Key on Apple iOS 13.3

**Cipher Text**



**Key2**

**Key1**

**Key1**

**Cipher Text**



**Encryption**

**K2**

**Cipher Text**

**Key3**



Key on Apple iOS 13.3

**Key2**

**( Picture2)**



**Key1**

**Key4**

**Key3**

**Key2**





**(Picture3)**

**Asymmetric Key Cryptography**

**Also known as public key cryptography**

We used here two different keys i.e

1:Private key

2:Public Key

Every Node has both Public and Private key which is used to encrypt or decrypt the message. If a system A uses its Private Key to encrypt the message then System B should use its Public key to encrypt it not its Private key

Both key will be used in this process.

**Private Key**

**Public Key**

**Public Key**

**Private Key**

Key on Apple iOS 13.3Key on Apple iOS 13.3Key on Apple iOS 13.3Key on Apple iOS 13.3Key on Apple iOS 13.3

**Private Key**

**Private Key**





**Encryption**

**Decryption**

**Using Public Key**

**Using Private Key**

**(Acceptable)**



**Encryption**

**Decryption**

**Using Private Key**

**Using Public Key**

**(Acceptable)**



**Encryption**

**Decryption**

**Using Public Key**

**Using Public Key**

**(Not Acceptable)**



**Encryption**

**Decryption**

**Using Private Key**

**Using Private Key**

(NOT ACCEPTABLE)

Lets assume system A wants to send message(transaction) to system B so how asymmetric cryptography make it secure?

Lets understand the whole concept of asymmetric cryptography.

Assume we have 5 **Nodes** in blockchain and every **Nodes** have its Public and Private key.

Public keys are open and every node know about the public keys of each other.

If **Node A** wants to send the message to **Node B** then **Node A** will used **Node B**’s public Key to encrypt the message and everyone can see the encrypted message but no one can decrypt it.

Only **NodeB** whose public key is used by **NodeA** can decrypt the message



**Encryption**

**Using B’s Public key**

**Cipher Text**



**B**

**A**

**Decryption**



**Using Only B’s Private key**



**Using D’s Public Key**

**Encryption**

**Cipher Text**

**C**

**D**

**Using D’s Private Key**

**Decryption**

