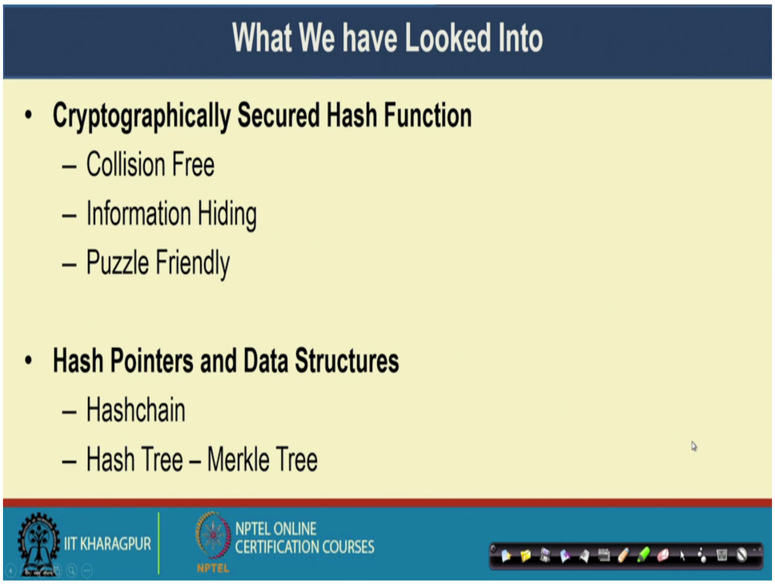
Blockchains Architecture, Design and Use Cases
Prof. Sandip Chakraborty
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture – 06
Basic Crypto Primitives – II

Welcome back to the course on blockchain technology. So, in the last class we have looked into different aspects of cryptographic hash functions and how do cryptographic hash function can be utilized for generating this secured and temp tamper proof series of blocks which works as the fundamentals behind the blockchain architecture.

Now, in today's lecture we will look into the details of another fundamentals behind secured blockchain architecture which is the digital signature. So, we look into the concept of digital signatures in details.

(Refer Slide Time: 00:59)

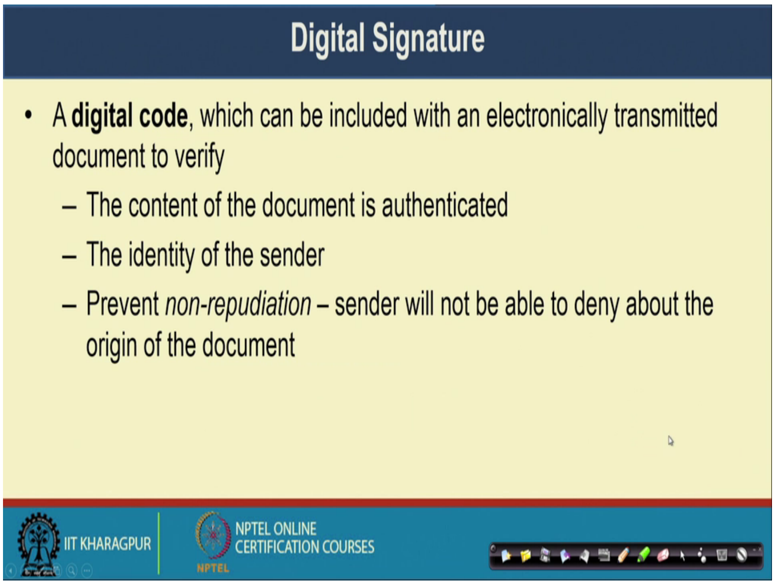


What We have Looked Into

- Cryptographically Secured Hash Function
  - Collision Free
  - Information Hiding
  - Puzzle Friendly
- Hash Pointers and Data Structures
  - Hashchain
  - Hash Tree – Merkle Tree

So, what we have looked till now that we have looked into the cryptographic hash function and it is properties like the collision free property, the information hiding property and the puzzle friendly property. And how these properties helps in development of the hash point or architecture and different kind of hash based data structure like a Hashchain and a Merkle tree. And by utilizing this concept of Hashchain and a Merkle tree how can you utilize it for the development of the entire blockchain architecture.

(Refer Slide Time: 01:33)



Digital Signature

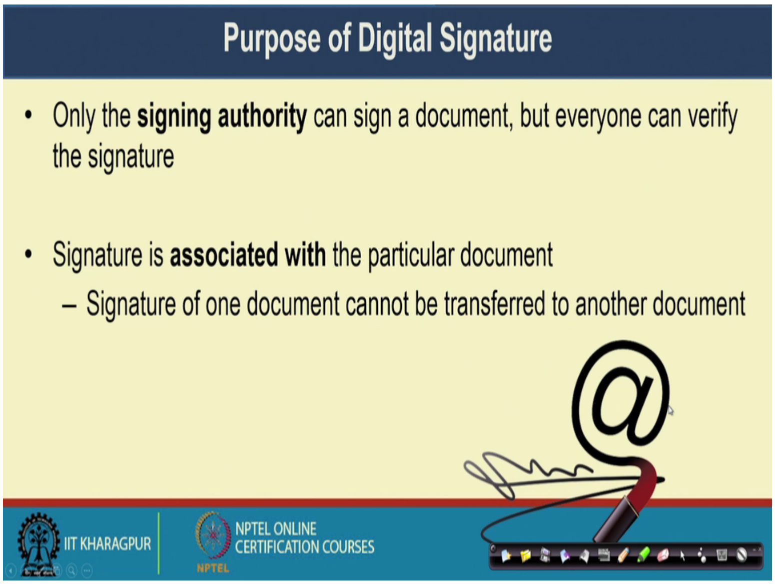
- A **digital code**, which can be included with an electronically transmitted document to verify
  - The content of the document is authenticated
  - The identity of the sender
  - Prevent *non-repudiation* – sender will not be able to deny about the origin of the document

Next we will look into this concept of digital signature. So, our digital signature is a digital code which can be included with an electronically transmitted document with this digital code you can verify first of all whether the content of the document is authenticated or not. So, it works like a sign say whenever you are sending a document or say a physical document from one person to another person in an administrative domain you need to physically sign the document and then send it which works as verification that the physical document is the authenticated document.

So, in the same way whenever we are transmitting some kind of digital content or digital file you put a electronically generated code with that particular file which will authenticate that file. Second this concept of digital signature it also helps you to identify the user like who has signed the document and the person who has signed the document that person should not be able to deny later that he or she has not been side sign the document or his or her signature should not be forced by another person.

So, it broadly prevents one kind of security attack which we called as non repudiation. So, this non repudiation attack is basically scenario a type of attack where the sender denies that he or she has signed this particular document. So, whenever we are preventing a non repudiation attack we say that the sender and the receiver will not be able to deny that he or she has signed the documents. So, it works as a verification of the origin of the document. So, this way the digital signature works pretty similar to a physical signature only like it is a digital entity which is transmitted with the electronically generated content or electronically generated document.

(Refer Slide Time: 03:40)



Purpose of Digital Signature

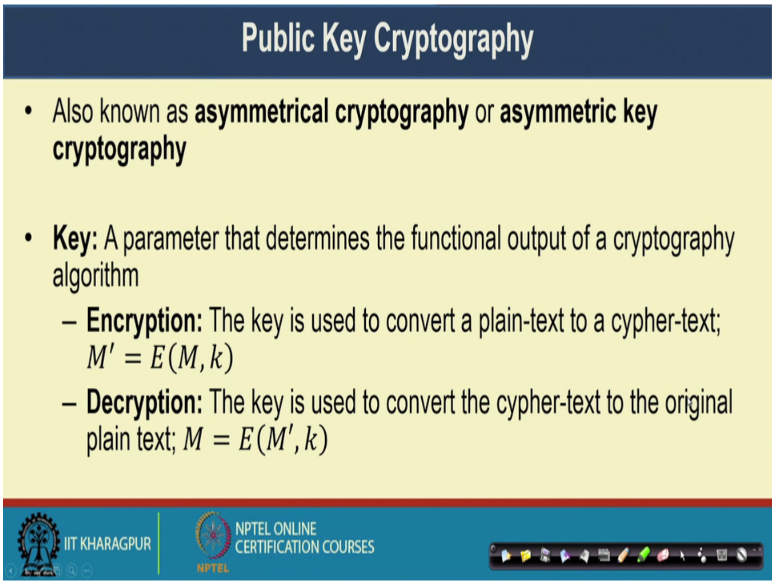
- Only the **signing authority** can sign a document, but everyone can verify the signature
- Signature is **associated with** the particular document
  - Signature of one document cannot be transferred to another document

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES

So, the purpose of the digital signature is that fast only the signing authority can sign a document, but everyone else will be able to verify the signature. So, the signing authority will be signing the document and no one else will be able to force that sign. So, the signing authority later on will not be able to deny that he or she has not been signed, but everyone else should be able to verify that the sign is originally signed by the intended authority.

Now, the signature it is also associated with a particular document which actually helps in proving the authenticity of the documents so; that means, like signature of one document that should not be transferred or that cannot be transferred from one document to another document. So, you cannot cut the signature from one document and paste it over another document. So, that way the digital signature should provide the authenticity of the document and at the same time it provides an identity of the origin of that particular document.

(Refer Slide Time: 04:52)



Public Key Cryptography

- Also known as **asymmetrical cryptography** or **asymmetric key cryptography**
- **Key:** A parameter that determines the functional output of a cryptography algorithm
  - **Encryption:** The key is used to convert a plain-text to a cypher-text; $M' = E(M, k)$
  - **Decryption:** The key is used to convert the cypher-text to the original plain text; $M = E(M', k)$

IIT KHARAGPUR NPTEL ONLINE CERTIFICATION COURSES NPTEL

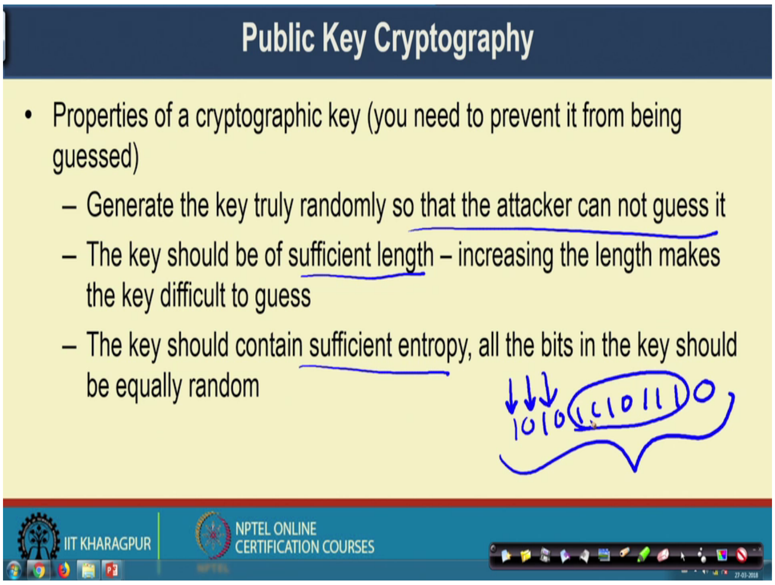
Now, this concept of digital signature we can realize it with the concept of public key cryptography, this concept of public key cryptography is an interesting concept in the cryptographic domain it is also known as asymmetric cryptography or sometime people call it as asymmetric key cryptography because we use two different keys. Now before going to the public key cryptography concept let us look into the concept of a key in the context of a basic cryptography.

So, key is a parameter which determines the functional output of an cryptographic algorithm. So, whenever you are applying an cryptographic algorithm the objective of the cryptography algorithm is to encrypt the particular information. So, that encryption of an particular information is done with the help of a key and again by using another key you can decrypt the information and recrypt back the original information. So, in this cryptography key there use for two different purposes for encrypting a document. So, the key which is used to convert the plain text to a encrypted text which we call as the cypher text.

So, here key work this k works like a key. So, this encryption algorithm it takes the input as the original information and an key as an input and produced the corresponding cypher text that the cryptography text corresponds to the original information that has been passed into and the decryption algorithm here we can again use the key either the same key or a different key, if we use the same key for decryption purpose we call it as the private key cryptography or the asymmetric key cryptography.

If we use different keys we call it as a public key cryptography or asymmetric key cryptography. So, in case of decryption algorithm you use either the same key or a different key which convert the cypher text or the cryptographically encrypted text to the original plain text; that means, you take this encrypted text as the input and the key as the input and finally, produce the original text.

(Refer Slide Time: 07:07)



Public Key Cryptography

- Properties of a cryptographic key (you need to prevent it from being guessed)
  - Generate the key truly randomly so that the attacker can not guess it
  - The key should be of sufficient length – increasing the length makes the key difficult to guess
  - The key should contain sufficient entropy, all the bits in the key should be equally random

So, the properties which should be there for a cryptographic key is that and this property is actually ensure that you are you need to prevent the key from being guessed by others, if the others are able to guess the key they will be able to encrypt the particular content or will be able to decrypt the particular content.

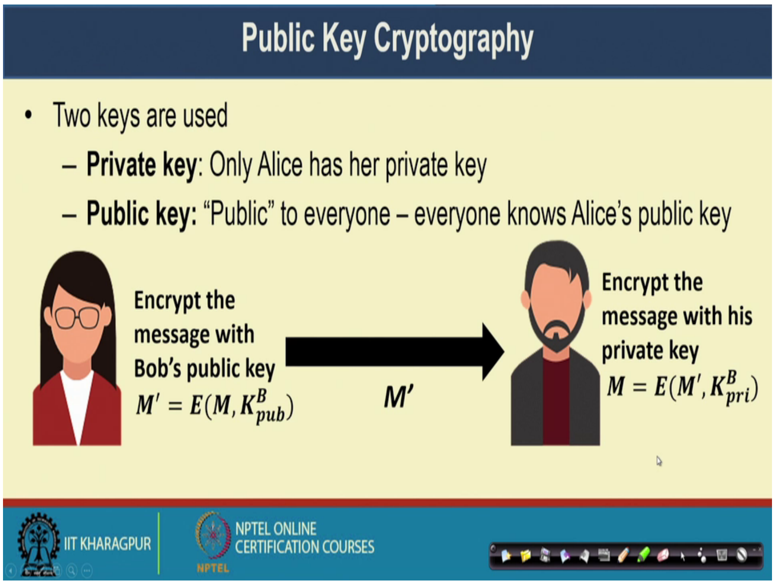
So, in the cryptographic purpose remember that the cryptographic algorithm is known to everyone what is known to what is not known to different person is the key. So, key is the secret thing which every person should possess and with the key it controls the output of your encryption or the decryption algorithm, if you provide the correct key then you will be able to decrypt and encrypt the data or you will be able to correctly encrypt the data otherwise you will not be able to do so.

So, the properties of a cryptographic key to prevent at the guess ability are as follows. So, first you need to generate the key truly randomly so, sorry. So, you need to generate the key truly randomly. So, that the attacker cannot guess it or no one will be able to guess that particular key unless that is informed explicitly to others, now the key should be of sufficient length. So, if the this length of a key has direct relation with your with your security of the encryption mechanism or encrypted text. So, if you are increase the length it make makes the key more difficult to guess. So, a 2 bit key will be easier to guess compare to a 128 bit key.

Now, the third property which is also important like the key should have sufficient entropy; that means, all bits in the key should be equally random. So, this concept of entropy basically determines that how much information is embedded inside a particular text or a particular information. Now, whenever you are generating some say random key you should be ensure that the entire thing is random or difficult to guess and at the same time every individual bit in the key they are also generated by a truly random mechanism so, they could not be guessable.

So, in case a attacker is able to guess say some partial bit pattern it becomes easier for the attacker to guess the entire key. So, that is why you need to preserve the entropy of a particular key; that means, a particular key should have sufficient entropy so, that all the bits in the key are equally random.

(Refer Slide Time: 09:56)



Public Key Cryptography

- Two keys are used
  - **Private key**: Only Alice has her private key
  - **Public key:** "Public" to everyone – everyone knows Alice's public key

Encrypt the message with Bob's public key $M' = E(M, K^B_{pub})$

M'

Encrypt the message with his private key $M = E(M', K^B_{pri})$

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES

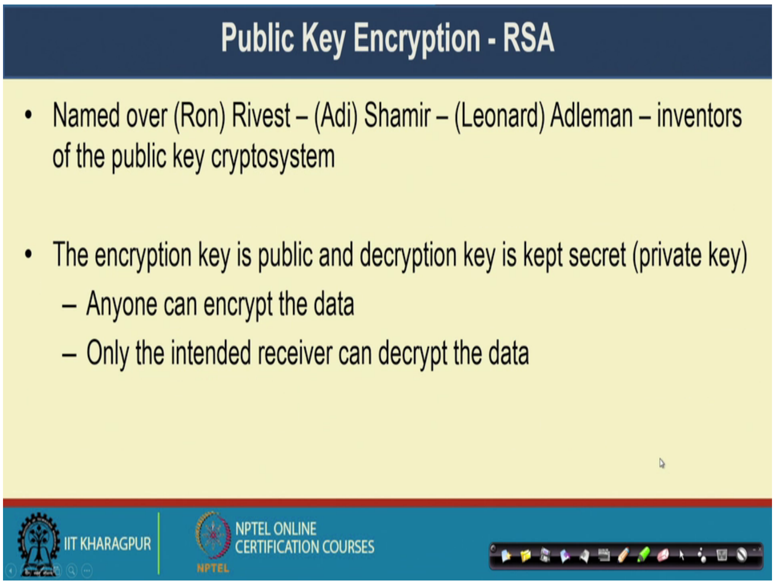
So, in case of public key cryptography as I have mentioned that we normally use two different keys one key is called the private key and the second key is called the public now in case of a encryption algorithm using this public key cryptography. So, the private key is the key which only Alice know so, Alice only has her public key. On the other hand Alice also possess a public key which is public to everyone; that means, although the key belongs to Alice, but everyone others in the universe they know that key.

So, whenever Alice is transferring some message to Bob in this case first Alice need to encrypt the message with bobs public key. So, remember that Bob’s public key is known to Alice, but Bobs private key is only known to Bob. So, that way the encryption is done with the help of public key because everyone knows that particular key.

Now, whenever the encrypted message is transferred now because bob only knows our private key. So, Bob will only be able to decrypt that particular message no one else in the universe will be able to decrypt that message. So, that was the intention of an encryption algorithm that whenever I am transferring some data to you during that time you will only be able to decrypt the message, but none in the universe will be able to decrypt that particular message.

So, that is ensured with the help of this kind of public key cryptography concept where the public key is used to encrypt the message whereas, the private key is used to decrypt the message. So, the destination note or the final note he or she only knows the private key information he or she only has the private key information. So, he or she will be able to decrypt the message with the help of that private key.

(Refer Slide Time: 11:54)



Public Key Encryption - RSA

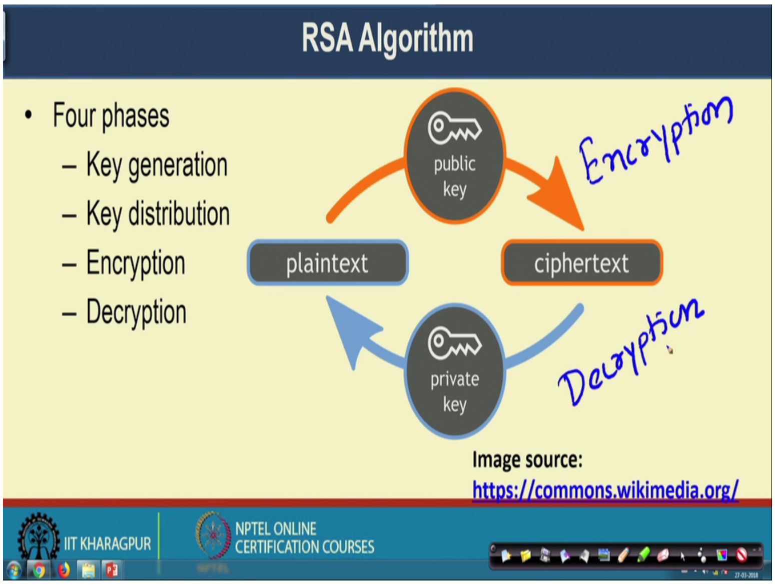
- Named over (Ron) Rivest – (Adi) Shamir – (Leonard) Adleman – inventors of the public key cryptosystem
- The encryption key is public and decryption key is kept secret (private key)
  - Anyone can encrypt the data
  - Only the intended receiver can decrypt the data

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
NPTEL

Now, let us look into a public key encryption algorithm although this particular algorithm is not used in concept of blockchain in blockchain or I will not say blockchain rather in bit coin purpose in blockchain you can use any kind of public key encryption algorithm based on your design. But bit coin uses a separate public key encryption algorithm, but we will use this particular RSA algorithm in our discussion because it was one of the fundamental public key algorithm, which was proposed initially who are proposed by the inventor of this public key cryptography concept and that particular public key cryptography algorithm is widely accepted in our community.

So, this RSA algorithm is actually named by the inventor of this algorithm Ron Rivest - Adi Shamir and Leonard Adleman. So, they together developed this public key encryption algorithm. So, R come from Rivest, S comes from Shamir and A comes from Adleman that way we have this RSA algorithm. So, in RSA algorithm the like a normal public key algorithm technic the encryption key is public and the decryption key is kept secret which is the private key. So, anyone can encrypt the data with the help of the public key and only the intended receiver he or she will be able to decrypt the information with the help of his or her private key.

(Refer Slide Time: 13:28)



RSA Algorithm

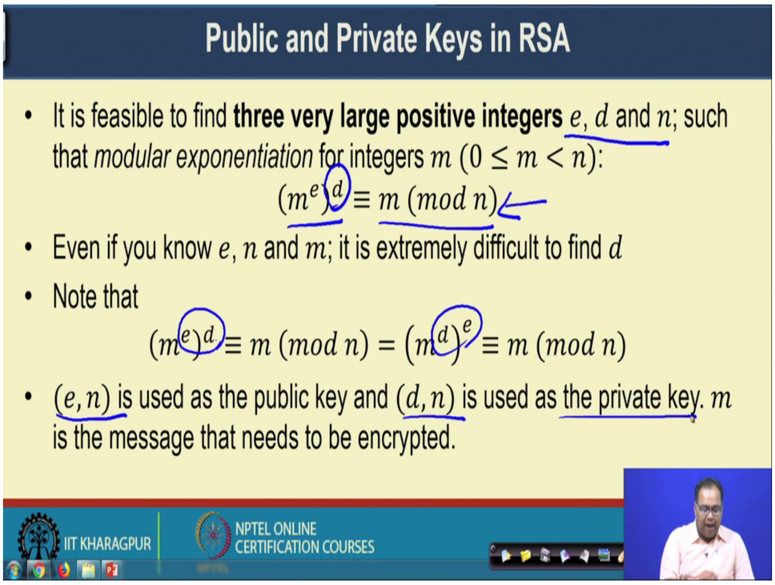
- Four phases
  - Key generation
  - Key distribution
  - Encryption
  - Decryption

Image source:
https://commons.wikimedia.org/

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES

So, this RSA algorithm works in 4 phases key generation, key distribution, encryption and decryption. So, as I have mentioned here that the public key is used to generate the cipher text from the plaintext and the private key is used to generate the plaintext from the cipher text. So, this is basically your encryption part and this is your decryption part.

(Refer Slide Time: 14:04)



Public and Private Keys in RSA

- It is feasible to find **three very large positive integers** $\underline{e, d \text{ and } n}$; such that *modular exponentiation* for integers m $(0 \leq m < n)$:

$$\underline{(m^e)^{\boxed{d}}} \equiv \underline{m\ (mod\ n)} \leftarrow$$

- Even if you know e, n and m; it is extremely difficult to find d
- Note that

$$(m^{\boxed{e}})^d \equiv m\ (mod\ n) = (m^{\boxed{d}})^e \equiv m\ (mod\ n)$$

- $\underline{(e, n)}$ is used as the public key and $\underline{(d, n)}$ is used as the private key. m is the message that needs to be encrypted.

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES

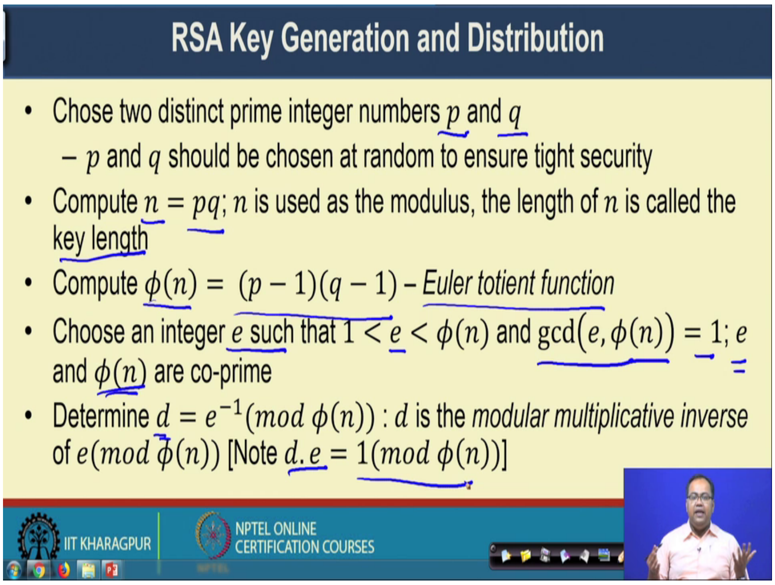
Now, let us look into that what is the mathematics behind the public and private keys in RSA. So, the fundamental mathematics we use in RSA algorithm to generate the keys in RSA algorithm is as follows that it is always feasible to find out three very large positive integers e, d and n such that the modular exponential for integer m. So, this modulo operation is the remainder after division operation. So, you can always find out e, d and n 3 such large integer such that m to the power e whole to the power d is functionally equivalent to m mod n. So, m mod n is second the mod n is the remainder of the division operation.

So, m to the power e, whole to the power d that will be functional equivalent to m mod n so, in this case even if you know e n or m 3 parameters of this particular operation it is always very difficult to guess the fourth parameter. So, that is comes from the discrete logarithmic problem where if you say know e m and n it is very difficult to find out this value of d.

Now, one interesting observation of this function is that if you do m e to the power d which is equivalent to m mod n which is further equivalent to m to the power d to the power e m mod n. So, you can you can change the order of this exponentiation and this helps you to generate two different keys the private key and public key. So, in this particular context e and n it is used as the public key and d and n it is used as the private key in case of RSA algorithm now if you have this information e d and n then you can encrypt a particular message at the same time you can decrypt a particular message. So, in case of a public key we use this e and n as the public key part and public key pair and this d and n are use as the private key pair.

Now, let us look inside the algorithm that how this RSA algorithm works.

(Refer Slide Time: 16:41)



RSA Key Generation and Distribution

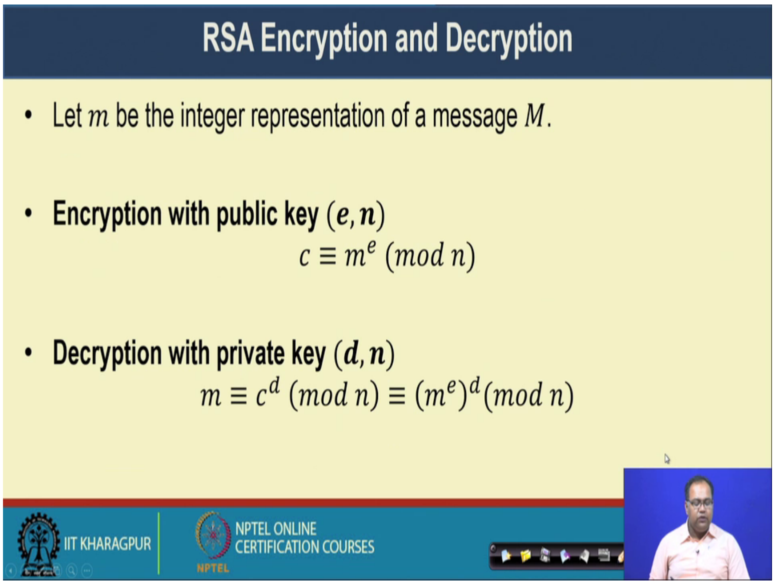
- Chose two distinct prime integer numbers p and q
  - p and q should be chosen at random to ensure tight security
- Compute $n = pq$; n is used as the modulus, the length of n is called the key length
- Compute $\phi(n) = (p-1)(q-1)$ – *Euler totient function*
- Choose an integer e such that $1 < e < \phi(n)$ and $\gcd(e,\phi(n)) = 1$; e and $\phi(n)$ are co-prime
- Determine $d = e^{-1}(mod\ \phi(n))$: d is the *modular multiplicative inverse* of $e(mod\ \phi(n))$ [Note $d.e = 1(mod\ \phi(n))$]

So, first we will look into the key generation part and how the key is distributed. So, for that we choose to distinct prime integers p and q remember p and q needs to be prime integers so, that it is difficult to guess such kind of integers the primality text is a difficult text. So, p and q should be chosen at random to ensure tight security of your algorithm. Now you compute the value n which is equal to p q so, n is used here as the modulus and the length of n is called the key length of RSA algorithm.

Now we call compute a function called phi n phi n is equal to p minus 1 into q minus 1. So, this particular function is known as Euler totient function apart from this Euler text totient function although the original RSA algorithm used this Euler totient function, but there are multiple other totient functions that you can use in the RSA algorithm in the place of phi n. So, here we are explaining the algorithm with this Euler totient function where phi n is equal to p minus 1 into q minus 1.

Now, you have to choose an integer e such that e belongs to 1 and phi n and gcd of e of phi n is equal to 1; that means, e and phi n these 2 integers are co - prime to each other. So, once you have found out this value of e then you determine the value of d which is e inverse mod phi n. So, here d is the modular multiplicative inverse of e of e mod phi n, now here you can see that d dot e d multiplied e it is equal to one mod phi n. So, that way you have the value of d e and n now you can generate the public and private key pairs.

(Refer Slide Time: 18:45)



RSA Encryption and Decryption

- Let m be the integer representation of a message M.
- **Encryption with public key** $(\boldsymbol{e}, \boldsymbol{n})$

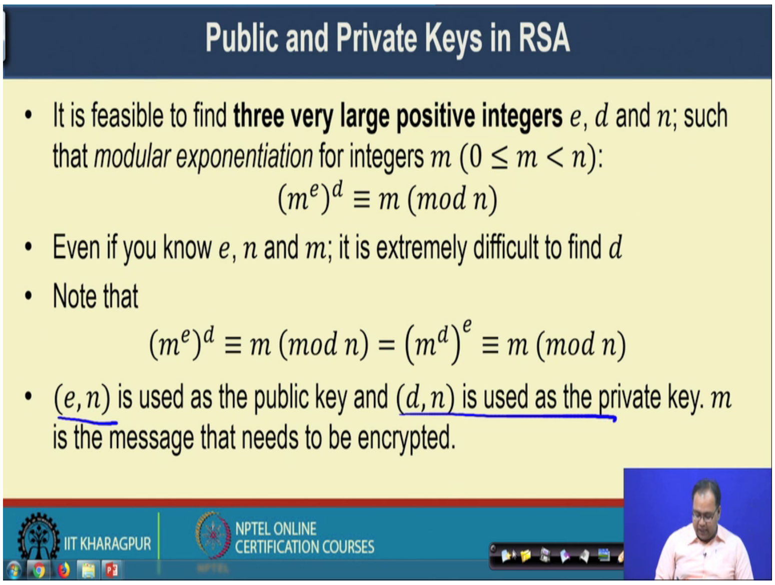
$$c \equiv m^e \ (mod\ n)$$

- **Decryption with private key** $(\boldsymbol{d}, \boldsymbol{n})$

$$m \equiv c^d \ (mod\ n) \equiv (m^e)^d (mod\ n)$$

So, once you have the private and the public key pairs now once you have generated your private key and the corresponding public key.

(Refer Slide Time: 18:59)



Public and Private Keys in RSA

- It is feasible to find **three very large positive integers** e, d and n; such that *modular exponentiation* for integers m $(0 \leq m < n)$:

$$(m^e)^d \equiv m \ (mod\ n)$$

- Even if you know e, n and m; it is extremely difficult to find d
- Note that

$$(m^e)^d \equiv m \ (mod\ n) = \left(m^d\right)^e \equiv m \ (mod\ n)$$

- (e, n) is used as the public key and (d, n) is used as the private key. m is the message that needs to be encrypted.

So, as you have mentioned earlier that your e and n it is used as the public key and d and n it is used as the private key. Now once you have generated the values the values the d e and n then you keep the private key with you and make the public key public to everyone. So, you publicly announced the public key, but do not announced the private key keep it with you.

Now, whenever you whenever someone else is going to transfer some message to you they can encrypt the message with your public key and send that message back to you once you receive that message, then after receiving that message you can decrypt the message with the help of your private key. So, that is the encryption and the decryption algorithm for this RSA mechanism.

So, let m is the message and this small m is the integer representation corresponds to this small m is the integer representation corresponds to this message. So, you encrypt with this public key e n whenever someone else is going to transfer the message with you and the encryption algorithm will be m to the power e mod n and whenever you are decrypting the original message. So, this is the encrypted message so, this is your cipher text. So, you transfer that cipher text to me and whenever I have received that cipher text after receiving the cipher text I can decrypt the cipher text at as c to the power d mod n.

Now, note that c to the power d mod n it is equal to m to the power e to the power d mod n and what we discussed earlier that yeah what we discussed earlier that we have generated the e and d in such a way. So, such that m such that m to the power e to the power d becomes equal to m mod n. So, in that sense whenever you are making c to the power d mod n it becomes m to the power e to the power d mod n which comes to be the original message m. So, that way you can decrypt the message by apply this private key d n.

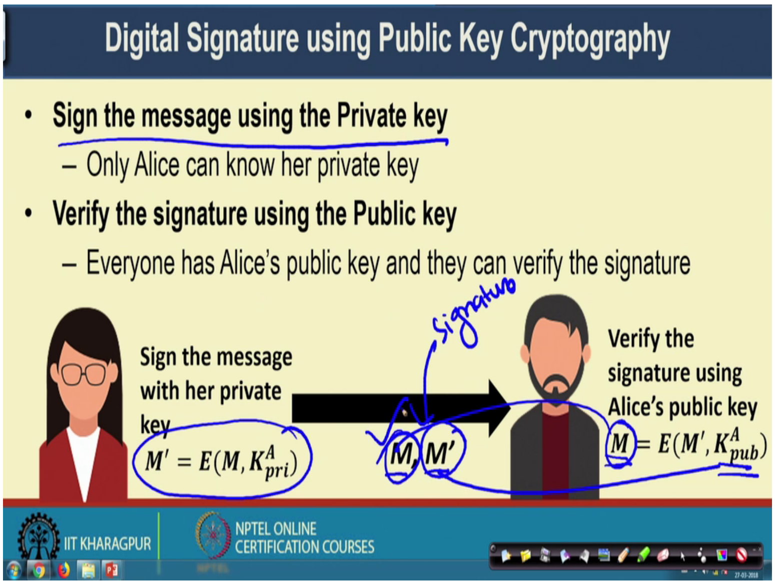
So, this way by applying the RSA algorithm you have a public key and a private key, whenever you are interested to transfer the message say whenever Alice wants to transfer some message to Bob Alice encrypt the message with Bobs public key and once bob received that cipher text then bob can decrypt the cipher text and get back the original plain text by applying his private key.

Now, we will look into that how we can apply this public key cryptographic concept in the digital signature purpose. So, that is a interesting concept because during the normal encryption we encrypt a message using a private we encrypt a message using a public key and then we decrypt the message with the help of a public key. Now, you know that the private key is only available to the intended person, but the public key is available to all with this concept what you can do that, if some person encrypt a particular message with his or her private key then it can work like a digital signature because you know that only in the universe every person has his or her own private key no one else and have the private key of Alice say Alice.

So, that way Alice has his Alice has her own private key with herself and if Alice encrypt something with her private key then everyone else can decrypt that message and check the originality of the message and can find out that the message has been encrypted with Alice private key and the private key was like a digital signature where by using the public key you are validating the digital signature whereas, the private key is used to sign the particular document.

So, that way in case of a digital signature you can use this concept of a public key cryptography where you sign the message using the private key because only Alice can know her private key and whenever you are verifying the message you verify the message using public key because everyone has Alice's public key and using that Alice's public key they can verify the message.

(Refer Slide Time: 23:58)



Digital Signature using Public Key Cryptography

- **Sign the message using the Private key**
  - Only Alice can know her private key
- **Verify the signature using the Public key**
  - Everyone has Alice's public key and they can verify the signature

Sign the message with her private key

$M' = E(M, K^A_{pri})$

M, M'

signature

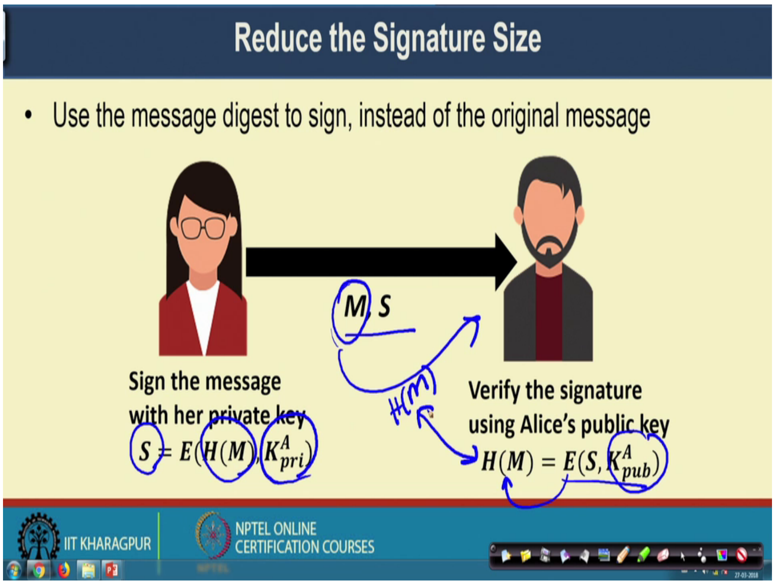
Verify the signature using Alice's public key

$M = E(M', K^A_{pub})$

So, whenever Alice is transferring some message to bob Alice signed the message with her private key using this public key cryptography and with the message she transferred sign. So, here M prime the cipher text works as a signature. So, that cipher text works as a signature here. So, you transfer the signature along with the original message.

Now, whenever Bob wants to verify it Bob can decrypt this message with Alice’s public key. So, K A pub is Alice’s public key. So, he can decrypt the message using Alice’s public key and then check whether the original plain text is returned that plain text is equal to this plain text or not. If this two plain text are equal; that means, the sign is valid the sign has been done by Alice because Alice can only possess her private key no, one else in the universe can possess Alice’s private key so, it works like a authentic proof of signatures.

(Refer Slide Time: 25:05)



Reduce the Signature Size

- Use the message digest to sign, instead of the original message

M, S

Sign the message with her private key

$S = E(H(M), K^A_{pri})$

Verify the signature using Alice’s public key

$H(M) = E(S, K^A_{pub})$

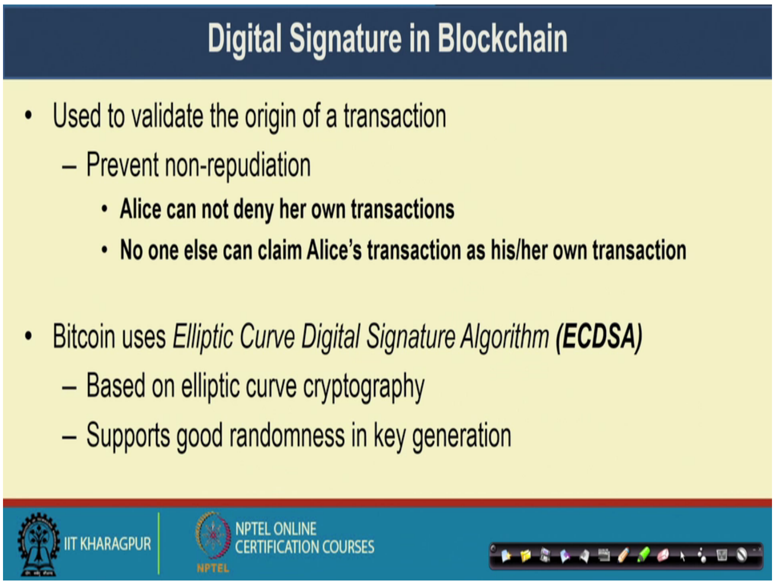
Now, by combining this concept of cryptographic hash and digital signature you can also reduce the size of the signature, say ideally if you are transferring say 1024 bit message with the 1024 message if you apply normal public key cryptography you will get a 1024 bit cipher text. Now, your signature is also 1024 bit long, if that is the case then your signature size is huge and in generally do not want a large signature. So, what we do that, we normally side on the message digest rather than on the original message. So, what you do that, whenever you are generating the signature during that time you first hash the message get the message digest.

So, Alice want to sign the message so, Alice first generate the hash of the message after generating the hash of the message then sign the message using the private key of Alice and generate the signature and whenever you are transferring the data along with the original data you also transfer the signature. Once Bob received this particular signature then Bob can verify it by decrypting the signature by applying the decryption algorithm, we using Alice's public key and then by applying this algorithm Bob will get the message digest and once Bob get the message digest the original message was also available to Bob.

So, from this M Bob can compute H M and can check whether these 2 H M matches with each other or not if these 2 H M and matching with each other; that means, the signature is correct and it provides you an authenticity of the original document. So, that way you are able to reduce the size of the signature by applying the message digest and at the same time you can support the authenticity of a particular document.

Now, let us look into that how we apply digital signature in Blockchain.

(Refer Slide Time: 27:13)



Digital Signature in Blockchain

- Used to validate the origin of a transaction
  - Prevent non-repudiation
    - **Alice can not deny her own transactions**
    - **No one else can claim Alice's transaction as his/her own transaction**
- Bitcoin uses *Elliptic Curve Digital Signature Algorithm* **(ECDSA)**
  - Based on elliptic curve cryptography
  - Supports good randomness in key generation

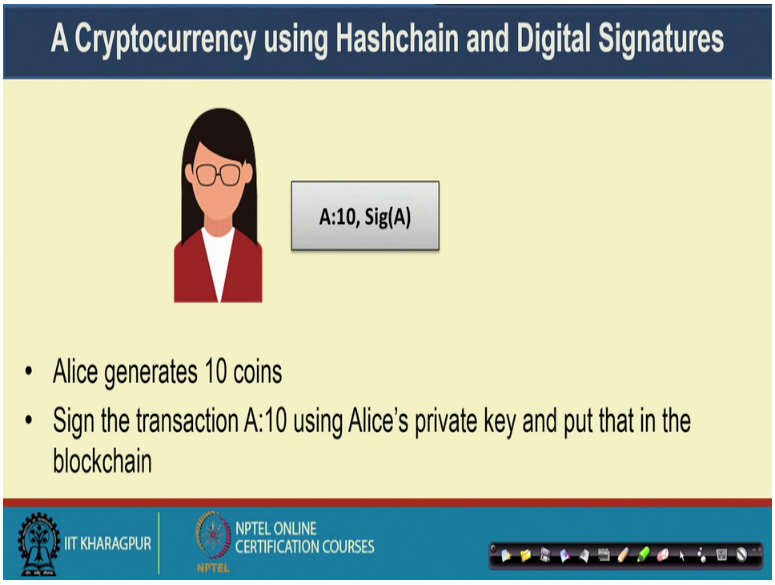
IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
NPTEL

So, this digital signature in a blockchain is used to validate the origin of a transaction to prevent the non repudiation kind of attack that I have men mentioned earlier. So, once Alice insert a transaction inside the blockchain, if that particular transaction is signed with Alice's a private key then later on Alice will not be able to deny that the message or the transaction has not been initiated by her.

So, that is one thing and the second thing is that no one else will be able to claim Alice's transaction as their own transaction because the transaction is associated with the signature the digital signature provided by Alice.

Now, Bitcoin it uses this elliptic curve digital signature algorithm or ECDSA algorithm it is another public key encryption algorithm based on elliptic curve cryptography which supports very good randomness in key generation. So, I am not going to explain the details of this EDCSA algorithm digital signature algorithm. So, if you are interested you can also look into that and explore how ECDSA algorithm generates the keys and how it is used to encrypt and decrypt a particular message.

(Refer Slide Time: 28:32)



A Cryptocurrency using Hashchain and Digital Signatures

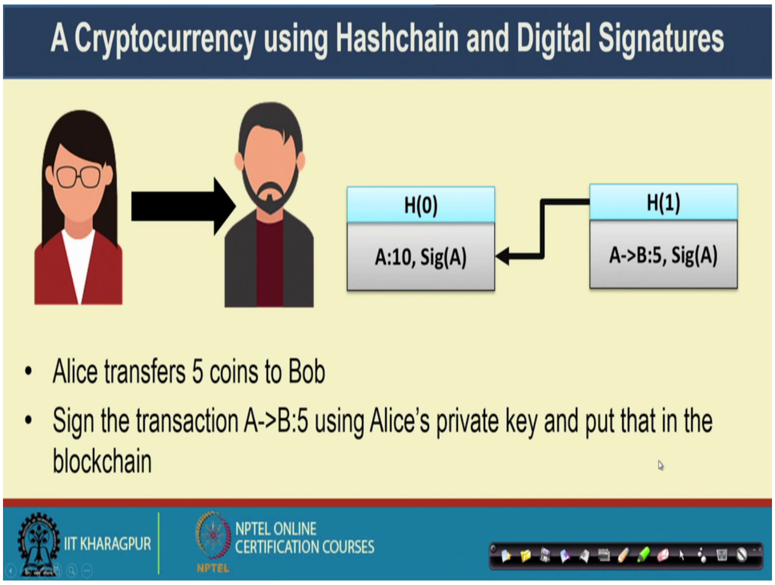
A:10, Sig(A)

- Alice generates 10 coins
- Sign the transaction A:10 using Alice's private key and put that in the blockchain

Now, let us look into a very simple Cryptocurrency using by combined this cryptography primitives that we have discuss the concept of Hashchain and the digital signature, say Alice wants to generate some coins. So, Alice generates some 10 coins. So, Alice transaction Alice has cert initiates a transaction that she has 10 coins with her.

So, with this particular transaction it puts her own signature so, that way Alice will never be able to claim that this transaction does not originate from her and no one else will be able to claim this transaction from Alice.

(Refer Slide Time: 29:14)



A Cryptocurrency using Hashchain and Digital Signatures

- Alice transfers 5 coins to Bob
- Sign the transaction A->B:5 using Alice's private key and put that in the blockchain

Then when Alice wants to make a transaction to bob what Alice does Alice make another transaction of say some 5 coins to Bob. So, Alice makes another transaction that from A to B 5 Bitcoin has been transferred and along with that she puts her own signature. So, whenever Alice's putting her own signature again Alice will not be able to deny later on that she has not transferred 5 bitcoin to Bob and no one else will be able to claim this particular transaction and we put all the transactions in the form of a blockchain using these hash pointer architecture where the hash pointers hash of 1 this particular hash value it will point to the hash of this first block.

So, it everything is put in a blockchain architecture by utilizing this hash pointer and you have this records of transactions along with the digital signature and once you have this records of transactions by utilizing that records of transactions, you can also always verify the old transactions that was the concept behind this blockchain. So, you put all the transaction in the blockchain along with the signature of the person who has initiated the transaction. So, that way you are making the blockchain tamper proof and at the same time you are ensuring that no one will be able to deny their own transactions or no one will be able to claim other others transactions.

So, the interesting problem here with the respect of cryptocurrency is that how will you maintain the economy. So, in case of your normal environment the bank they generates the new coins with time and they also deal it the old coins at the old money from the system, but in a digital environment where everyone is generating their own money and everyone is making the transaction so, one after another in a decentralized way. So, you in that particular bitcoin architecture if you take it as an example there you do not have a central authority like a bank who will create or destroy the coins with respect to time.

So, the question comes the important question comes that how can you bring this kind of economic aspects under the notion of digital currency. So, that the coins are generated automatically with time or the coins are destroyed automatically with time. So, that is another interesting concept that will discuss in the in our next few lectures while will talk about the details of this bitcoin architecture.

So, in this particular lecture we have given you an broad overview about the basic cryptographic primitives like the cryptographic hash function and the digital signatures which works as the fundamental building block behind this entire blockchain architecture as well as the bitcoin fundamentals. So, in the next classes we will discuss about the architecture of bitcoins and how we utilize bitcoin to maintain a digital currency ecosystem where in a complete decentralized way people can maintain their own currency and they can make the transactions over the universe. So, see you all during the next classes.

Thank you all.