

Introduction to Blockchain Technology and Applications
Prof. Sandeep Shukla
Department of Computer Science and Engineering
Indian Institute of Technology-Kanpur

Lecture No. 06
Blockchain Technology and Applications

Welcome back. So we were discussing the Bitcoin consensus problem. And we ended up with discussing the impossibility result FLP. And we said that that is a worst case scenario. And also there are a number that is for deterministic case and there are no match algorithms for Byzantine fault tolerance. But in case of Bitcoin, the consensus works slightly differently than all existing Byzantine fault tolerant algorithms. And actually it works quite well in practice.

And the theory is, not exactly in line with the practice. And but theory is important because you might not; you might think that it is working all the time. So I am fine. But theory gives you the impossibility results are the lower bounds, which can tell you whether you have not thought about some corner case when your system will not work or not proceed, you may not be able to decide on the next block and things like that.

So but we will see now; how this thing works in bitcoin and then you will see how we actually circumvent the problem. So, first of all, as we said before that many times if you change the model of the problem, so in case of a distributed database consensus, for example, which in which context the FLP was thought about, the model does not have a certain elements.

(Refer Slide Time: 01:53)

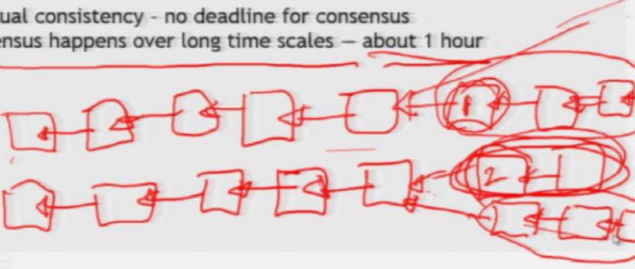
Some things Bitcoin does differently

Introduces incentives

- Possible only because it's a currency!

Embraces randomness

- Eventual consistency - no deadline for consensus
- Consensus happens over long time scales – about 1 hour



For example it assumes that all the nodes are either going to do the same thing deterministically. And some of them will be differently, but it does not provide in the model a way to make this malicious nodes to behave correctly by adding incentives. So, in Bitcoin, we add the incentives or rewards for behaving correctly. And if you have that, then your model changes and when your model changes, the existing results may also change because your assumptions have changed.

So, in the original Byzantine fault tolerant problem, there was no incentive. So, we were not saying that if you behave maliciously, then you will not get somebody's word. But if you did not behave maliciously, we will give you rewards. So, if the if you introduce that, it is a different problem actually, also, instead of, you know, talking about deterministic algorithm you actually embrace randomness in case of bitcoin consensus.

And we will see how that works. But first of all, in the other consensus, the traditional consensus, we assume that this thing will go in through the protocol will go in around, and then it will eventually terminate. And at that point consensus will be achieved. And then again, another proposal will be made, and another round of consensus protocol will start, and then when it terminates, we will have consensus.

Here we actually relax that when we say that there is no deadline for consensus, we did not say that at this point, the consensus process ended. And this at this point, all the nodes should have

added the same block as the last latest block to their copy of the blockchain. We actually say that eventually be when you look at in the long run, be all the block chains replicas should be same, but we are not saying that at every round of consensus at the end, everybody will have the same replica.

So, this is called eventual consistency. And this is another relaxation from the original problem, because we are not saying that, you know, it has to be round by round consensus. So, in fact, in Bitcoin, the consensus happens over a long time scale, in this case, you know, about an hour. But even then about after an hour, the probability of that the different replicas are different becomes extremely low, but it is not deterministic.

And then if you wait for 10 hours, then the probability becomes close to almost negligible. And when we say this, this is how we say this right so, let is say we have this blockchain up to this and another replica also have this. They have the copy of the blockchain. And until then, they have this and until then let is assume that they are identical. Now, another block has been proposed, actually multiple blocks have been proposed.

And this guy adds one block. And this guy adds a different block. So 1, block 2 and at this point, I did not have the consensus exactly worked out because they decided on 2 different blocks. But we will see over time, one of these, let is say this guy, after 1 hour will have, let is say, 6 more blocks on top of it, and this guy will have only people, some people built the next block on top of this, but other people said this does not look right.

So, I built I will build on top of this, and then this chain keeps growing, then what we are saying is that at the end, this will become consistent. So, this will become what we call an orphan part of the chain, and the chain will proceed here like this, whereas this copy has always been in the right path. Now, it depending on what 1 and 2 are, it could be that this guy, this guy will grow up and this guy will eventually converge to this this way.

So, at this point, we did not know, but over time over 1 hour when 6 more blocks have been put on top of this one, by then you if it still survives, the one survives and 2 does not have 6 blocks

built on top of this and then there is Alternative Testing, then they will, they will have a one has become permanent. And eventually this guy will have this chain replicated here. So that is the idea of eventual consistency.

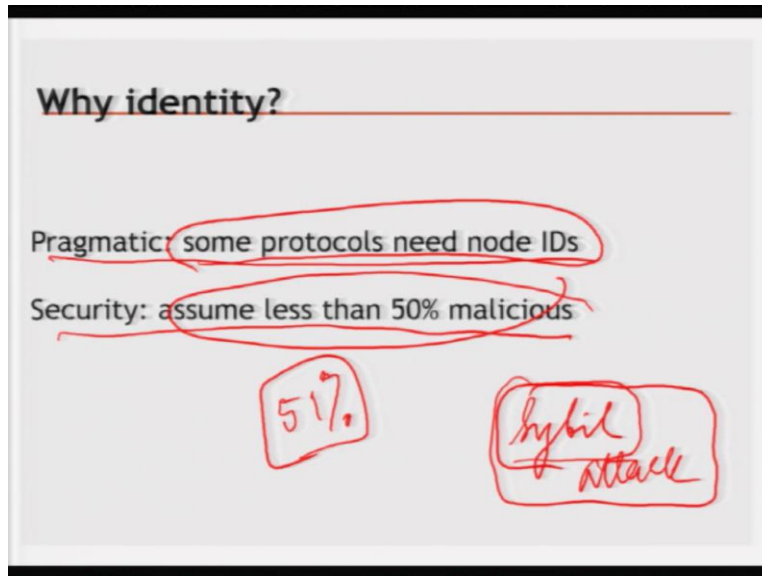
(Refer Slide Time: 07:16)



The other issue is that here we are doing consensus without an identity we did not know who the people are so we did not know, we only know their hash of their public key as their account address. And that is all we know about them. And it may be that multiple people, multiple accounts belong to the same person. So if you are thinking about consensus, but if I have, if I create 10,000 accounts, but I am the owner of all 10,000 accounts, I may actually think that I have more power to overwhelm the consensus.

But especially if I am a malicious node, and I want to burden the conscience consensus then I can create that many nodes.

(Refer Slide Time: 08:05)



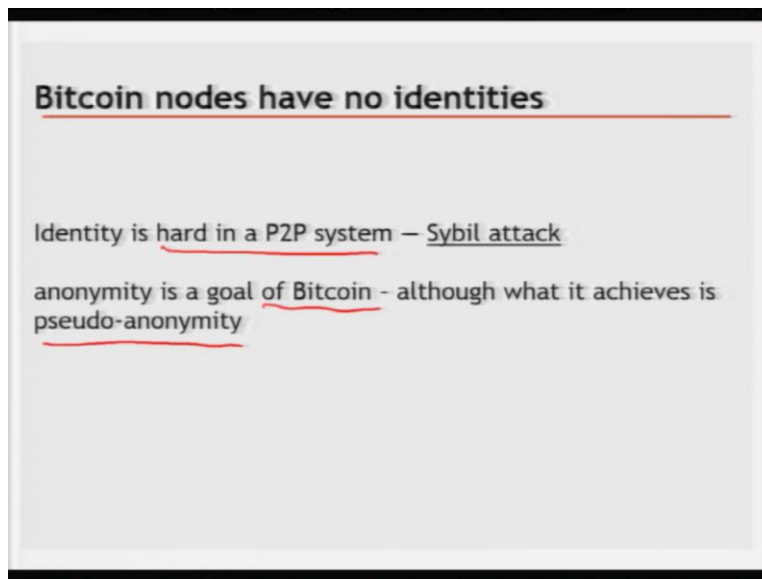
So that when you create multiple aliases for yourself and nobody knows that it is your alias, because there is no way to know and then you get some advantage there is called a Sybil attack. So your protocol should also be robust or resilient to Sybil attack also, right? Because otherwise, you know, somebody who has more malicious intent will create more ideas and then he will have more, let is say voting power, for example.

So therefore, we have to actually make sure that the protocol is resilient to Sybil attack. Now, some other protocols require node IDs. And maybe also they connect that to real world identity. For example, we will see when we will look at permission blockchain in permission blockchain every node is actually permissioned. And how are they permissioned? Because there is some way of somebody who actually permitted them.

So there is a central centralization there that who permits them. But after the permission process is done, the rest of the stuff becomes decentralized. But in that case, we know the identity of each node, the real world identity or some other identity, in this case in bitcoin and aetherium is called permission less, and we did not know the real identity, and somebody may have multiple identities. In fact, we saw yesterday that the change address, for example, requires that you create your own multiple identities.

So, under these circumstances, we have to make some assumptions. So we assume that we already talked about this that we have to always assume how many of them are malicious, we cannot say 100% could be malicious because then the protocol would not work. So we assume that less than 50% are malicious. So in that assumption, under which the Bitcoin protocol works, if anybody gets 51% malicious, then and these malicious nodes actually collude with each other, then the protocol would not work.

(Refer Slide Time: 10:15)



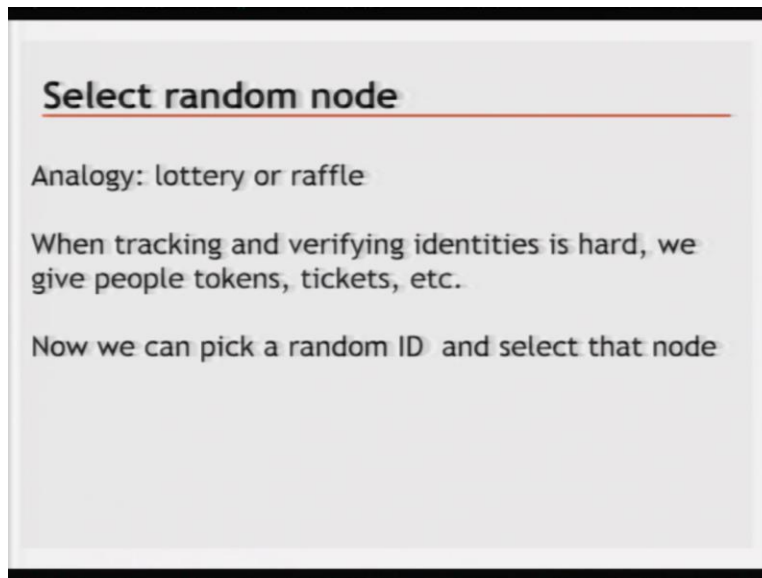
So, as I said that in a P2P system permission free permission less so, we cannot really determine who and when we are making the P2P system in such a way that there is a low barrier to entry. So, anybody can create a public private key pair and enter the system. In that case, Sybil attack may happen. On the other hand, the goal of Bitcoin creation was also not only decentralization, but also give anonymous it.

Now, in the very beginning that this anonymous it is not real anonymity. It is actually pseudo anonymity because people are using various data mining techniques because all the data of a blockchain Bitcoin blockchain are publicly available. So, you can actually look at the correlation between different addresses, how they transact with each other, and so on. And they can actually determine that these addresses seem to be from the same entity and, things like that.

So it is in a sense that it is pseudo anonymous, but, as I said that it is almost impossible to establish the real identity even if you do all this to know which accounts are together and which accounts are not together, but you cannot really still know who the real person behind that identity is so but that is a goal of the bitcoin. Now, we saw that this goal also has some downsides, which is the use of bitcoin addresses as ransomware addresses.

The use of bitcoin in the drug trade and another dark wave activity, so that is a different issue here by anonymity is the reason why identity is not there, which makes the consensus even more challenging.

(Refer Slide Time: 12:05)



So the way the consensus works, mimics the following process. Suppose there is a raffle like you go to a concert. And then at the end of the concert, there is a raffle. And they will give away a couple of prizes by picking your name from some kind of hat or some box. Now, they did not know your names, they did not know your identity. So what they do is that they look to take at the top of the ticket that you bought, so it has the same ticket number.

And then the put the ticket number in the box, these stubs of the ticket in a in a box and pick one randomly, and then say, whoever holds this wins. So in there, your real identity is not required. You are given randomly, some number, random ID, and then I am picking one of them. So suppose you want to discuss that way I will do the consensus is that everybody proposes their

blocks. And then I will give everybody a random ID. And then I will pick one of the random ID, and then say that your block will be the next block.

So this is what we want to do but, again, we did not have a central authority to do this if there was a central authority, then the central authority can arbitrarily decide whose block will be taken central authority could have done a raffle kind of thing to pick them, but central authority is not there. So we have to actually do it in a decentralized fashion, we would want to do this, but we want to do this in a decentralized fashion.

(Refer Slide Time: 13:41)

Key idea: implicit consensus

In each round, random node is picked

This node proposes the next block in the chain

Other nodes implicitly accept/reject this block

- by either extending it
- or ignoring it and extending chain from earlier block

Every block contains hash of the block it extends

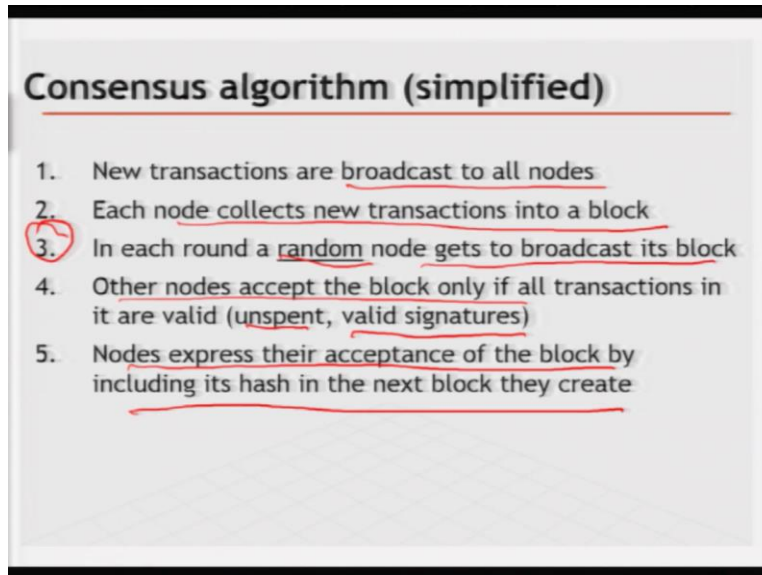
The diagram shows a sequence of blocks represented by squares. A central block has an arrow pointing to the right, indicating it is the next block to be proposed. Another block to the left has an arrow pointing to it, showing it is being extended. A third block below the central one also has an arrow pointing to it, showing an alternative extension path.

The key idea is that in each round, a random node is to be picked. And this node will be allowed to do give the next block to the chain. And other nodes will then implicitly accept or reject this block by either extending it so when you have a new block to the existing chain. The others may decide that they will extend your block or they might say, I did not think that this block should be there. So I will go to the previous block and extend from that. And by extend, I mean that when I create a new block, I put the hash of one of the previous blocks.

So I can put the hash of this guy, or I can put the head hash of this guy and then conceptually, if I do the hash of this guy here, then I am connected to this and basically I am extending this part of the chain from here. And if I do this, then I am extending the chain from here. So that is how I say all the other nodes will say that I accept you as the latest block. So every block decides every

new block that is created will either extend the latest new block or it will extend the previous block and that accordingly the chain will grow. So this is how the system grows.

(Refer Slide Time: 15:04)

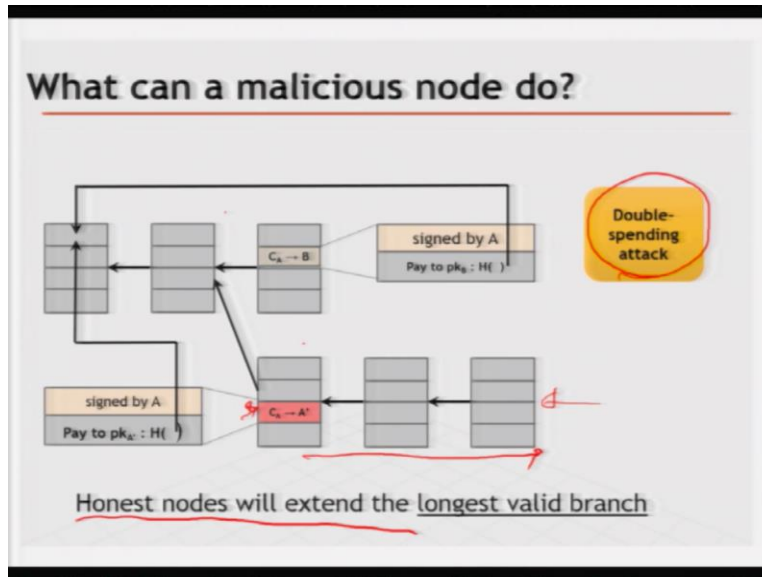


So a simplified form of how this works is that all new transactions are broadcast to all the nodes through the peer to peer network. Each node collects new transactions and decides to create a block. And as we discussed that every node may have collect a different set of transactions into a block, and the blocks are fixed size. And in each round, a random node gets to broadcast its block. Now here we are, we are not saying the whole story because which random note gets to broadcast is blocked is something that we will leave it for later.

All the other nodes accept the block only if all its transactions are valid. Now if you have sent a block in which there are invalid transactions, then no node will accept it except for maybe malicious nodes, which is that money that you are stealing has not been spent before that is you are not doing double spending. And they have to check whether the signature is valid, then they have and this is they do for all transactions in the block.

And they say that, this block is valid. Now, the nodes then will express their acceptance of the block by including its hash in the next block they create. So, that is the overall scheme of things. Now this step is what we have to discuss more because how is this random node gets selected, because we have no central authority, so we have a decentralized system.

(Refer Slide Time: 16:39)



So before that, let us see what a malicious node can do. So let us say I have the blockchain until this and then I make a transaction which makes into the next block. So I am giving some money to be and this is C_A is the coin that I have right now. And I sign it so it is a valid transaction. Now, what you can do is that you can try a double spend so, you say, give this money to my other address. So, A might decide that. So B let us say a merchant.

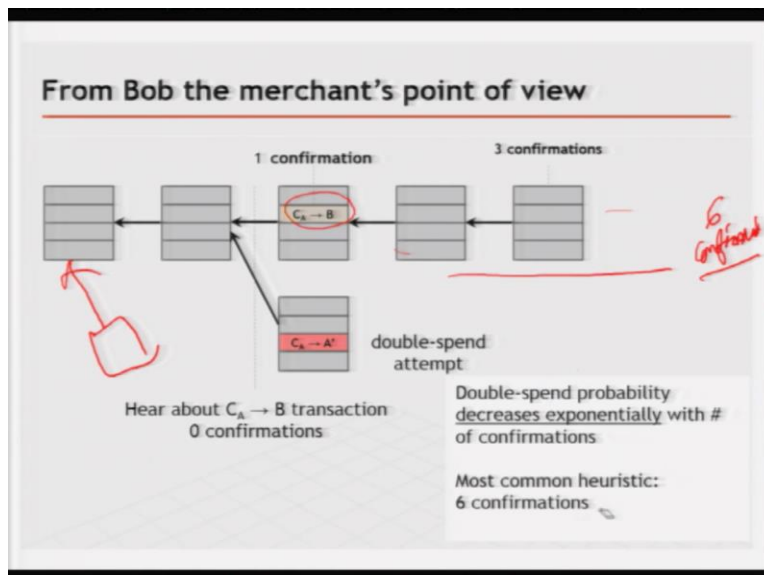
And you say to be that send me this package, I am going to buy and I am going to pay you in Bitcoin. And the way you know that I have paid in Bitcoin, you listen to the broadcast, and very soon you will see the transaction where I am saying that I am paying you. And then also, if you wait a little more, you will see that in the next block, if the merchant believes after seeing the next block that he has been paid, and ships the merchandise.

At this point, what A can do is that once he gets acknowledgment that the merchandise has been shipped, he can actually then put another transaction and then get, which pays the same money to himself in another alias. And then make sure that all his friends and others are built the other blocks from the block that contains this second transaction. So then this transaction, this block becomes an orphan block, if somehow this block becomes extended.

And that is a double spending attack. So an attacker, in this case A might actually then in the meantime, the merchandise has already been shipped. So the merchant after a while the merchant goes and checks the block blockchain again and sees that this has become orphaned in the current latest blockchain does not contain that transaction. And therefore he cannot spend the money he got because his transaction is not there in the actual blockchain.

So these doubles spend and attack is a big problem. So an honest node is supposed to extend the longest valid branch. So, if this becomes a valid branch see if this double spend has can be considered a valid branch because this branch has become orphan. So, this is the honest nodes may start building from here and therefore depriving B of his payment.

(Refer Slide Time: 19:20)



So, the merchant's point of view this is really a loss because he is not getting any confirmation. So, what we say is when the block mix when somebody makes the transaction part of the block and that block becomes part of the block chain, and then we say it is a single confirmation. Then when somebody builds another block on top of this block, then we say we get another confirmation and then we get another block built on the same chain then we say the third confirm confirmation.

So, if the confirmations keep coming on the original one, then Bob will be okay because after enough number of confirmations, this block will become orphan and so therefore this attempt to

double spend will be getting no confirmation. But in case there is their confirmations keep building. So normally in Bitcoin until 6 confirmations, you did not ship your merchandise. So that is and we will see later that it takes about 10 minutes for one new block to be mined.

So, 6 confirmations will take 60 minutes. So that is why we say that you have to wait until 1 hour see that the transaction that you are interested in, the block which has 6 confirmations, then you might more or less sure that people cannot really create a side chain, which will grow longer because then another 1 hour has to pass before this can be done, but honest nodes will not build because this is a smaller chain. So they will start building on top of this.

And therefore this chain will keep growing. So this is all probabilistic because you are assuming the honest nodes listen to, they actually abide by the protocols. You are also assuming that nobody equates later or 51% power, or 51% nodes did not become malicious and subvert the chain by changing from here itself, like they can start building here, all kinds of stuff can happen, but the probability is very, low.

So eventually, even after 6 confirmations, there is a very, trivial probability of being subverted. But we live with that and normally it works. So, double spend probability decreases exponentially with the number of confirmations, and this can be proven mathematically, because the process of new block creation usually follows a probability distribution and therefore inter block arrival time is an exponential distribution. And so most commonly heuristic is to wait for 6 confirmation so which is closely.

(Refer Slide Time: 22:04)

Recap

Protection against invalid transactions is cryptographic, but enforced by consensus

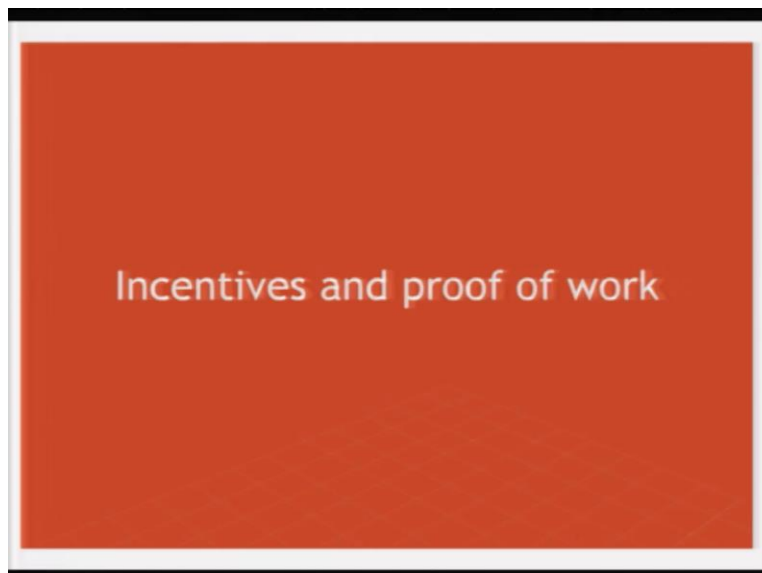
Protection against double-spending is purely by consensus

You're never 100% sure a transaction is in consensus branch. Guarantee is probabilistic

So to recap, the protection against invalid transaction is completely cryptographic, enforced by the consensus. Protection against double spending is purely by consensus. So when you check for transactions validity, the consensus does not come into play. If a transaction is invalid, every node will see that by checking the signature and whether the money belongs to you and things like that. And that can be done all with cryptographic techniques.

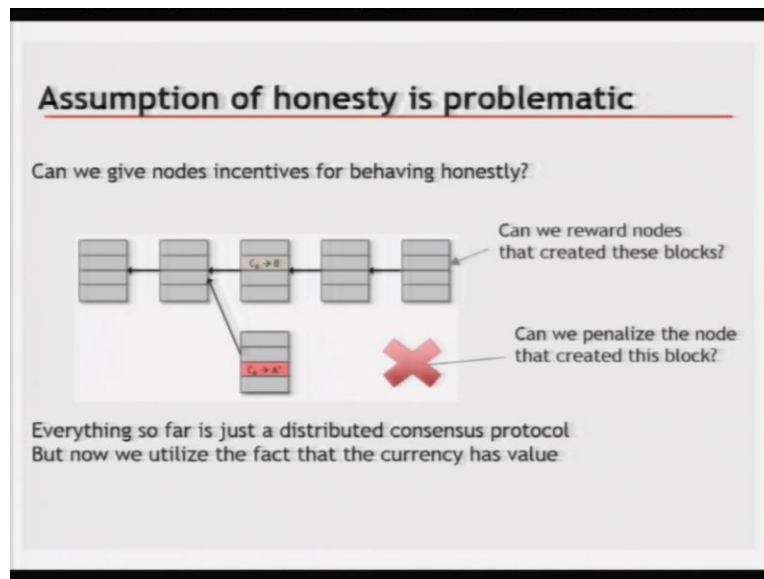
Protection against double spending is actually has to be done through consensus and it is probabilistic. And you are never 100% sure that your transaction is in the consensus branch. The guarantee is probabilistic, but with overwhelming probability.

(Refer Slide Time: 22:51)



So that is what we have seen so far. And now we talk about the incentives. So incentives are actually important part of the problem of consensus in blockchain. So we talked about this earlier, that unlike the classic system of consensus, distributed computing, Byzantine fault tolerant computing, there, we did not try to change the malicious nodes behavior towards behaving correctly. But in here, we actually do that by giving incentives. So let us look at the incentive mechanism that was put in place for this.

(Refer Slide Time: 23:32)



So can we give nodes incentives for behaving or behaving honestly, or can we penalize the nodes that behave dishonestly? Now, if there was a centralized system, somebody would be actually determining, this behavior is not acceptable, this behavior is acceptable. So we should penalize, but there is no central agent or central authority to do that, and therefore we cannot do this directly. So we have to have some mechanism for incentive. And those who are not behaving correctly will not get the incentive. So that is how it works.

(Refer Slide Time: 24:14)

Incentive 1: block reward

Creator of block gets to

- include special coin-creation transaction in the block
- choose recipient address of this transaction

Value is fixed: currently 12.5 BTC, halves every 210,000 blocks.

Block creator gets to “collect” the reward only if the block ends up on long-term consensus branch!

So therefore, what we were going to do is that we are going to show you 2 incentives. One is that when your block gets added, you can add a special coin creation transaction in the block and put the recipient address yourself. So whoever creates a block, he assumes that his block will be accepted. So in that block, he adds an extra transaction, which we call a coin creation transaction, which creates new coins. And then he says puts the recipient addresses one of his addresses.

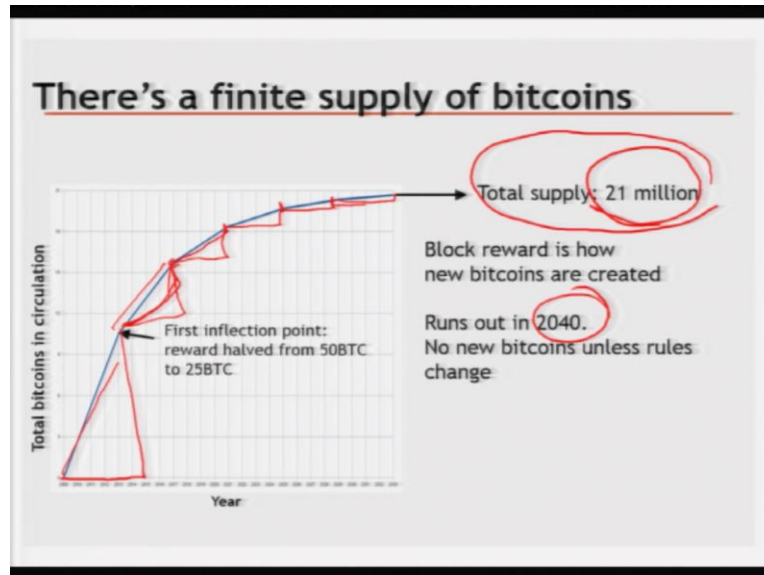
He has many aliases, he will put one of these addresses. Now how much he can create, but the coin creation transaction is fixed. Now it is a 12.5 bit points, but it has every 210,000 blocks. And so far with the 10 minute in interval between blocks, it is roughly about, I think 4 years. So therefore, right now if I create a block, I will put a transaction saying that create 12.5 new bitcoins and give it to my address.

Now; if I make block gets accepted, then that transaction becomes part of the blockchain. And then after 6 more blocks are built on that block you I get to use that 0.5 blockchain if my block does not get added to the to the blockchain, or if it gets added, but nobody builds on top of it and I did not have 6 confirmations, then I cannot use that. Those block chains that I created even though I put it in the block.

So the block creator gets to collect the reward only if the block ends up on long term contract branch the branch that has many, others built on top of it. So, at least 6 blocks has to be built on

top of that, in order for this to be able to use that 12.5 Bitcoin earlier it used to be 25 Bitcoin and in the beginning it was 50 Bitcoin and later on it will be 6.75 Bitcoin and so on.

(Refer Slide Time: 26:19)



So, and there is a finite supply of bitcoins. So, the only way to create Bitcoin is by this process your block has to be accepted, you will get reward and this way every 10 minutes somebody or the other will get some rewards and then more coins will be created. Now, in the beginning, as I said they do 350 per transaction, so the slope was higher of the total number of Bitcoin in the system. Then it became 25.

So the slope was lower, and then now it is 12.5 the slope was even lower, but you see the number of the time required to reach that 210,000 blocks is shrinking. So, this is 210,000 blocks were created in this more time here it is much smaller and this to 210,000 blocks require a lot longer I think. So, therefore, these things and then it will become 3.7 to 5 or something and, so on. So, eventually it will taper out 2040 I then the total number of bitcoins will become 21 million.

And beyond that no more bitcoins will be created unless the rules change. So, this provides the Bitcoin ecosystem some scarcity right because if you have a finite supply of bitcoins, then the prices will not be high right the exchange rates will not be high. So, therefore, in order to build in this scarcity, they have decided in the beginning that there will be only 21 million bitcoins

created. So, we will talk about the other incentives in the next session. At this point we have learned how to do one type of incentive for, nodes who play by the protocols, rules.