

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

Lecture – 44
Blockchain Security – I (Overview)

Hello everyone, welcome to the next lecture in our Blockchains Architecture, Design and Use Cases course. We have gone through a good bit of background material on what block chains are some permission block chain systems, hyper ledger fabric, hyper ledger composer and so on. We have also looked at a lot of use cases in a number of industries. I think it is now time to look at some advanced material alright. So, on these are all advanced material that are at the cutting edge of research and innovation today.

So, I am glad you all stuck on with this course until so long. So, let us look at some of this material and this all these might also be good starting point for you to for maybe some of your projects, your thesis even. So, I am hoping this is this is helpful for you. So, specifically over the next four lecture lectures we are going to take a deep dive into security and privacy aspects of blockchain systems. This first lecture is going to give you an overview of security and privacy, of some of the requirements of enterprise applications; what enterprise applications have on blockchain platforms.

(Refer Slide Time: 01:19)

Open Network: Security Properties

<u>Identity</u> (what defines system participants)	<u>Transactions</u> (network messages)	<u>Transaction Validation</u> ("correctness" of network messages)	<u>Transaction Ordering</u> (protocols to order transactions)
---	---	--	--

NPTEL

Security:


- Correct transaction validation
- Ledger immutability

Privacy:

- Pseudonymity, in some cases anonymity

Assumptions:

- > 50% computing power complies with protocol
- User wallet is safely maintained
- All contracts are deterministic (Bitcoin and Ethereum achieve this by restricting set of permissible operations)



"Attack the assumptions" & .. human error

- Compromise user-wallets by
 - Attacking online wallet services
- Compromise ledger immutability by
 - Motivating alternate behavior
 - Attacks shown at even ~25% compute power
- Exploiting smart contract vulnerabilities to arbitrarily change ownership of coins

Hard forks are sometimes inevitable...

3

So, let us just look at an open network. So, this is like the bitcoin kind of network, above permissionless network where there is a certain notion of transactions, where the transactions are all just exchange of a Bitcoins from one person to another. These are back transaction validation which is in the bitcoin case it is just prevention of double spending, making sure it is an authorized entity who is trying to spend bitcoins that they actually own.

The security is really in the fact that you want to make sure the transactions are all valid, before they are committed on to the ledger and once it is committed on to the ledger you want immutability. You want to ensure that no single participant is able to unilaterally modify the contents of the ledger. They cannot manipulate other entities to also convince them to manipulate the ledger. So, there are certain guarantees that are provided. So, in the bitcoin case a certain fraction of the nodes as long as they are not compromised you can still provide this immutability guarantee.

The privacy on the other hand the only kind of privacy that they provide is through the notion of pseudo anonymity, in some cases although there are permissionless networks that have full anonymity. So, we do we will talk about that later as in one of the lectures and what are some of the assumptions. The assumptions to provide some of these guarantees in terms of security and privacy are that not more than 50 percent of the computing nodes have been compromised right ok.

So, they are all at least 50 percent of the nodes are following the protocol as stipulated. This is not compromised, they are not malicious and they are all functioning as expected. It also assumes that the user wallet with the private keys of the individuals are all safely maintained, they are not compromised either. And, the fact whatever transactions you are running on blockchain they also have to be deterministic.

So, bitcoin and Ethereum achieve this by reducing the set of operations you can perform, but there are other ways to achieve this as well right. Certain these sort of assumptions may not always hold true and there have been many attacks that have been shown on some of these open networks. So, for instance one of the most common cases is where either the user wallets or the decentralized exchanges have been compromised in some way. And, users' secret keys have been stolen and bitcoins have been stolen because, of that I am using bitcoins generically here.

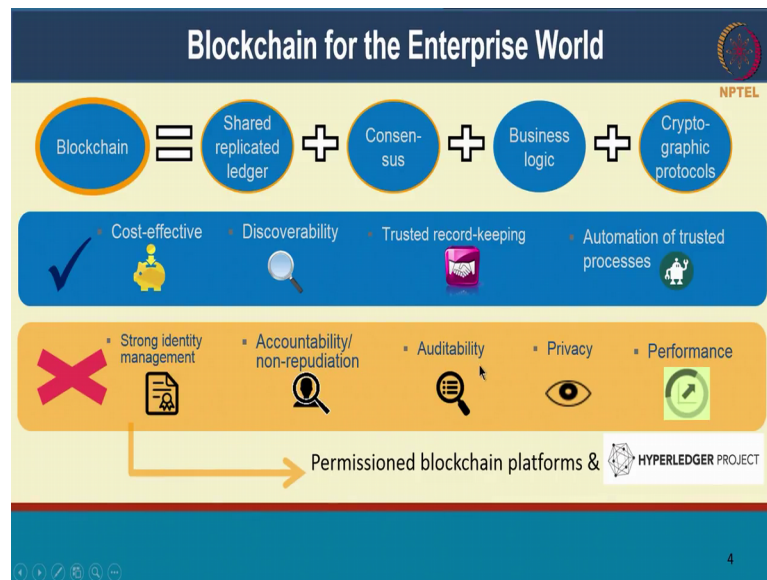
But there are other crypto currencies that have also faced a similar kind of issues and the ledger immutability can be compromised, if nodes behave in a certain different way. So, for instance there is a research article that shows that even with if even if you compromised just maybe even 25 percent of the network, you can actually make you can cause bad things to happen in the network ok.

So, that is the notion of selfish mining I will, but I will give you a reference to that later in this in this towards in the fund reading section. The other kinds of attacks for the ones that support smart contracts, people try to explore exploit vulnerabilities in the smart contracts; what one might think is bugs in the smart contract. And, that will allow the hacker to arbitrarily change the ownership of coins, the coins maintained by the smart contract itself right.

So, these are all exploiting vulnerabilities in the application code or the smart contract code, not really in the platform itself. The platform might be secured, but the way you have written your smart contract is not secured. And, one of the problems that platforms such as Ethereum have is that, Ethereum purports that code is law right. There is no governing body there is no legal recourse if something goes wrong in your code.

So, that will see the situations where hackers have hacked into the system stolen coins. And because, there is no legal framework to actually prove that this is theft and actually do anything about it; what ended up happening was there were hard folks and people actually had to undo transactions on the blocking. So, that actually goes against the principles of blockchain itself, it is supposed to be immutable and not subject to people modifying it by themselves. But it has happened that people have performed hard folks on the Ethereum blockchain for instance.

(Refer Slide Time: 06:11)

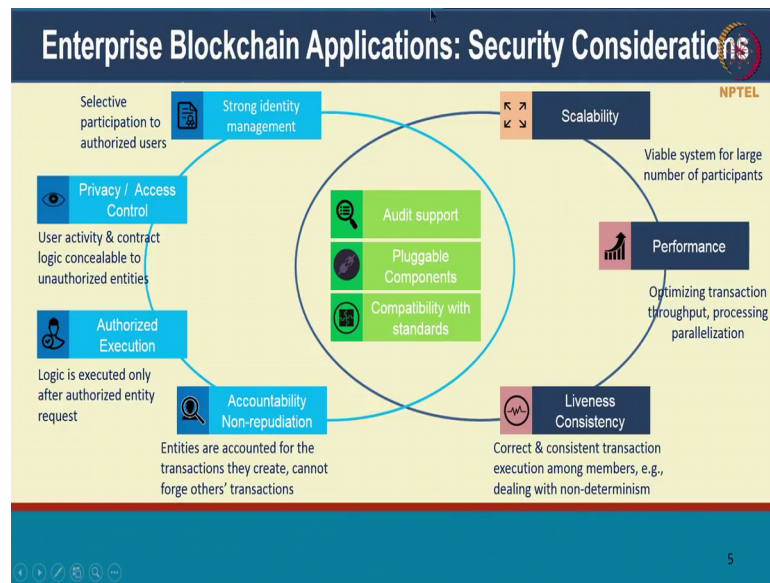


So, in the enterprise world apart from the fact that we have a shared replicated ledger, we want consensus. We do not want any one party having central authority, we want the ability for smart contracts to execute business logic and we want all of this protected through cryptographic protocols. So, apart from these properties there are certain other properties that are not provided in permissionless area. For instance, permissionless networks do not give you a strong identity management, they do not give you accountability non repudiation they do not give you the ability to audit transactions.

So, I should be able to know an auditor should be able to come in and find out who exactly did what and more importantly there is no notion of privacy. So, we will have one dedicated lecture speaking of different notions of privacy and how there are different capabilities that are there in the security literature; that that provides some of these privacy guarantees. And, privacy is also possible through design hyperledger fabric has some of those concepts, R3's corda.

We will talk about corda in a few lectures; R3 corda also has some of these privacy aspects. So, we will talk about what those are and over across all of these we want to make sure that the performance does not degrade while, we are providing some of these capabilities ok.

(Refer Slide Time: 07:38)



So, let us go over some of these security considerations. We talked about strong identity management, this is to say that we want to ensure that only a permission set of entities gain access to the network itself; others do not even know that this network exists or cannot read what is on the network. So, the participation into the network is authorized and governed. So, that is the strong identity management and this is typical of enterprise systems because, any enterprise knows that it is only going to do business with a certain set of other enterprises. And, it at least knows of those enterprises. You are not going to be doing business completely with unknown entities almost most likely, although you know those entities it does not mean that you trust them alright. So, those are two distinct things.

Then there is privacy and access control. So, for whatever information whatever transactions you are performing on blockchain, you want to make sure that only certain authorized entities are able to obtain access to that information. And, they are also only certain set of entities are allowed to perform certain transactions certain types of transactions on the blockchain right. Then there is authorized execution. So, this is to say that only certain set of entities should be allowed to see what the business logic is, see what the smart contracts are. May not be that all parties in the network see all the smart contracts that are part of it.

There might be different subsets of entities who need to know certain pieces of logic and they are the ones executing that logic. And, the next one is around accountability and non repudiation. So, once information is there in the system we should be able to prove clearly that this information was in fact, produced by this person. And, then once with that with that proof whoever performed that transaction should not be able to deny that they actually perform the transaction. And, one person should not be this the corollary of this is also say one person should not be able to forge out of the transaction to make it seem as though it came from somebody else rather than themselves right.

So, that is the accountability and non repudiation aspect and we want all of these properties together with support for auditability. So, we should be able to have regulators or auditors come in and see everything in the network or whatever selective portions you want either will reveal to them. We want this to be pluggable components. So, in hyper ledger fabric almost every component is designed as a pluggable component. So, the ledger is pluggable. So, you can have different aspects of the ledger, we have CouchDB and LevelDB and then you can have consensus as a pluggable concept.

Just like that we want all the security concepts should also be pluggable. Like the identity management should be pluggable, the crypto protocols should be pluggable and we will see how that is achieved in hyperledger fabric. And, the reason is also that some of these security protocols also matured with time rights, over a period of few years these protocols keep changing and the best practices keep changing. And, we want fabric to be able to adapt to some of those changes and improvements in security literature.

And, like I said we were all of these security properties with minimal impact to scalability performance consistency right. So, we do not want to compromise too much on the scalability and performance. Scalability is in the number of participants how much the network can grow in size, performance is in this throughput right. How many transactions you can handle per second or what is how quickly can you commit transactions; like what is the latency. So, those would be performance metrics we do not want to affect them, while providing much richer security and privacy guarantees.

(Refer Slide Time: 11:29)

Security and Privacy: Key Differentiation of Fabric

- Existing security/privacy controls
 - Membership and access control
 - Endorsement policies
 - Application-level encryption
 - Channels
 - SideDB
 - Trusted chaincode execution (secure containers)
- New security/privacy controls
 - Anonymous and unlinkable transactions (Identity Mixer)

NPTEL

6

So, security and privacy is key differentiation for fabric compared to many other platforms that are out there. And, fabric right from the beginning has been designed keeping security and privacy as the foremost code and that is also some of the things we are seeing from client's right. Security and privacy are the biggest things that clients ask of us when we are building fabric or when we are building applications on top of the fabric.

So, there are many controls that are provided in fabric to give you different notions of security and privacy, there is membership and access control. So, at every stage of fabric you can have access control. So, you can say who are the ones were part of participating in a channel, within a channel who is executing chain code, within the chain code what are the endorsement policies right. Who needs to sign off on a particular transaction so, that is really the endorsement policy.

You can apply application level encryption to provide privacy guarantees on who sees gets to see the data. So, unless you have the application layer key, the secret key that was used to encrypt the transaction you would not be able to decrypt it. So, then there is notions of channels there is SideDB. So, we will in one of the lectures actually in this in this week itself one of the lectures we will talk specifically about SideDB, as a construct for providing data privacy. So, we will talk about that and there are also hardware

capabilities like trusted chain code execution using inside secure containers and apart from these there are new concepts that are also coming in.

So, at the time of the recording there is a new capability using identity mixer for providing anonymous and unlinkable transactions. So, we will cover that as well, it is going to be part of fabric very soon. But it is not yet that, it is still in development it is an open source; you can see the development in progress, but it is it is still not merged with the master branch of fabric. So, so this is just a pictorial view of how at every stage there is there is some level of security and privacy embedded in terms of membership access control, the endorsement policies and the commotion of channel SideDB and so on.

(Refer Slide Time: 13:53)

Security in Cloud / Hardware

NPTEL

- IBM Blockchain Platform: All components run inside Secure Services Container

Protected Memory (SSC LPAR)

Diagram illustrating the boot sequence:

- Firmware Bootloader (with mSPIBios check and mDecryption) checks the Software Bootloader.
- Software Bootloader (with mDecryption Key and LUKS) opens the Software Image & Data (with Disk Encryption and LVM).
- Encrypted Software and Encrypted Data are shown below the bootloaders.

Complete isolation and encryption of code and data

Boot sequence

1. Firmware bootloader is loaded in memory
2. Firmware loads the software bootloader from disk
 - Check integrity of software bootloader
 - Decrypt software bootloader
3. Software bootloader activate encrypted disks
 - Key stored in software bootloader (encrypted)
 - Encryption/decryption done in flight when accessing appliance code and data
4. Appliance designed to be managed by remote APIs only
 - REST APIs to configure Linux and apps
 - No ssh (allowed in dev mode)

7

And as I mentioned there so, there is also apart from the protocol or the platform itself giving you security properties; we can also obtain security properties through cloud services or through hardware right. In IBM cloud, the IBM blockchain platform all the components of your network actually run inside secure services, inside what is called a secure services container. So, these secure services container is an isolated container in term in pretty much everything right.

In terms of memory, CPU, in terms of disk it is everything is isolated and there are some additional properties we will talk about them. So, your pair order chain code all of them execute in separate containers, they are all isolated with each other from one another. And, that prevents any kind of tampering that one organization might do to what might

try and impose on another ok. So, what is the secure services container do? So, it is actually a combination of hardware and software. It has a firmware bootloader, that is first loaded into memory and once you have that loaded into memory it then loads a software bootloader from disk.

And, when it loads the software bootloader from disk it can actually do an integrity check. This is to make sure that when what you read from disk was what was originally put inside. So, you can make sure the disk has not been tampered with and you can actually do that integrity check and you can decrypt the software bootloader. So, this is actually stored encrypted on disk and the software bootloader it actually has keys stored in there as key stored in the software bootloader, that will enable you to activate encrypted disks. And, this encryption decryption is done in flight when accessing appliance code and data that is residing on the on the disks right. So, that is the kind of security that is all built in into hardware and software as part of this secure services container. And, the application is designed so, that it can interact with the container only through APIs.

So, there is no concept of being able to ssh into, the machine there is no admin user; even IBM will not be able to get into this container and look at what is inside. So, when you develop an application, when you are run a network all of that is completely isolated and even IBM will not know what is running inside. So, you have complete control over that container that is provision for you right. The only thing IBM can probably do is shut down your container right, but it is not in IBMs interest or any cloud providers interest to actually do that. So, guarantee that is provided here is that even we will not have the keys and we will not have the ability to get into your container and look at the data or the code that is running inside.

(Refer Slide Time: 16:49)

Intel Software Guard Extensions (SGX)

- Trusted computing base
 - SGX Hardware (silicon chip + CPU microcode)
 - Code running inside the enclave
- Isolation of user-level code, protected from processes running at higher privilege levels
- Remote attestation of the enclave
- Reverse sandbox for applications

The diagram illustrates the SGX Protection Model. It shows a stack of layers: 'Protected Mode' (grey) and 'OS attack' (green). An 'App' (blue) is shown with an 'Enclave' (blue) inside it. A 'Malicious App' (red) is shown with 'Bad Code' (red) that is blocked from reaching the enclave. An 'Attack' (red) is shown being blocked by the enclave. An 'OK' arrow points from the enclave to the app. The diagram is labeled 'SGX Protection Model' and 'NPTEL'.

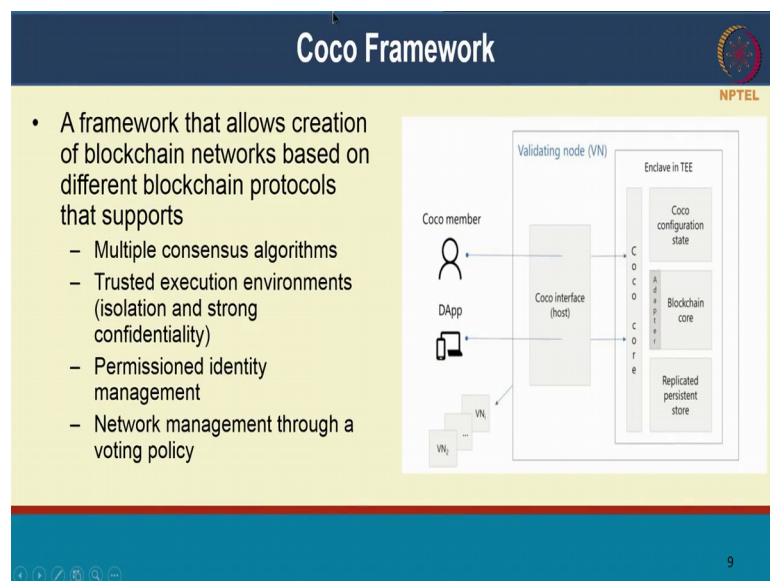
The Intel Software Guard Extensions: SGX is a as a incredible innovation, that Intel has come out with. It has just has similar features to the secure services container, but it is a bit different. So, what it has is within your application you can actually have enclave and this enclave is usually a small piece of code that is trusted. And, it is been verified and it is known to be correct and bug free and all of that. And, you can ensure that this enclave is protected from other processes that might be running on the same machine. Like for instance, this picture here in the bottom you have an application here, you want to protect a portion of that application or the whole application itself from other malicious applications, that might be running on the same operating system.

So, in the in the in the cloud when you are using virtual machines, it is possible that multiple applications are running on the same operating system or the same infrastructure. So, you want to prevent other malicious application to be able to attack and obtain information from my application that I want to keep secure. So, they have this notion of an enclave and this is a this is in hardware, it is a silicon chip plus CPU microcode. And, there is code that is running inside there that enclave and the code is basically isolated user level code. And, that is protected from other processes that are simultaneously running on that on that system, on where at even at higher privilege levels.

Then apart from that you can on the enclave code itself you can do remote at the attestation. So, what does that what; that means, is you can have a remote authority centralized authority, verifying whether this code when it comes up has not been tampered with. So, it is actually running the original piece of code that was running there. So, that is the attestation aspect and this actually provides a record reverse sandbox. So, there is within our application there is a sandbox for you or to protect certain piece of code or data in data in that object.

And, there are certain limitations to this, but I will not go into that. But, this has caused a good amount of innovation in this space the Intel SGX capability. And, today almost a lot of our laptops that are coming out today they are all come with SGX capability. So, if you dig into your laptop you will see that its most likely has SGX in them of course, if you have an Intel chip on your on your laptop.

(Refer Slide Time: 19:27)



So, this next one is about it is a now switching gears a different framework, is the framework of that's come out, Microsoft has come out with called the Coco framework. I am again going to talk about this very briefly because, this is an overview lecture. So, what this framework allows you to do is you to create block chain networks based on different blockchain protocols right. So, for instance you can have multiple consensus algorithms that you can plug in to your blockchain network. So, this big picture actually helps this is actually from the Coco framework white paper itself.

So, if you think about it so, what they allow you to do is it have they give you a pluggable way of plugging in your favorite suction platform, whatever that is. You can plug it that plug that into Coco framework, you can plug in your own consensus algorithm into that into that framework for that platform. And, it also has the persistent store right. So, it maintains the ledger for you. So, there is a blockchain core, this is the pluggable component and you will have to write an adapter for your favorite blockchain application; you can plug that into Coco framework and it will manage the consensus and the transaction execution and so on. And, apart from that it gives you permissioned identity management and it also uses trusted execution environments.

So, each of these things so, if you see this whole thing is running inside and enclave it is a trusted execution environment. This is actually they have a they are doing this with Intel SGX and they also provide you capabilities for network management; once you have this set up you can do network management through your voting policy. So, there is more like governance that fabric provides, but a version of that for just providing how you can would on your network management.

So, for instance you can vote on whether a new organization should join your network or not. So, it gives you an interesting set of capabilities basically for management and governance of your blockchain network, agnostic of the kind of blockchain platform that you want to use. So, you can just have an adapter and bring an any kind of blockchain platform to it.

(Refer Slide Time: 21:47)



Fun Reading

- “Majority is not enough: Bitcoin mining is vulnerable”, I Eyal, EG Sirer, International Conference on Financial Cryptography, 2014. Available at: <https://www.cs.cornell.edu/~ie53/publications/btcProcFC.pdf>
- “A Survey on Security and Privacy Issues in Bitcoin”, M Conti, S Kumar, C Lal, S Ruj: <https://arxiv.org/pdf/1706.00916>
- Understanding the DAO Attack, Coindesk blog: <https://www.coindesk.com/understanding-dao-hack-journalists/>
- Intel SGX Details: <https://software.intel.com/en-us/sgx/details>
- Coco framework whitepaper: [https://github.com/Azure/coco-framework/blob/master/docs/Coco Framework whitepaper.pdf](https://github.com/Azure/coco-framework/blob/master/docs/Coco%20Framework%20whitepaper.pdf)

10

So, that brings us to the end of the lectures a bunch of fun reading for you to go through. So, this first one is a research paper titled “Majority is not enough: bitcoin mining is vulnerable”. So, this is the selfish mining that I talked about. So, it shows how maybe even at about 25 percent network capacity, if you are able to take over maybe 25 percent of the network you can actually do very bad things into the in the network. So, that is that is this paper, it is a very interesting read I would encourage you to read it. It is one of the hot papers that is been cited a lot of times already and there is now “A Survey on Security and Privacy Issues in bitcoin. So, this is right now as a preprint available on archive.

So, you can go read that I think it gives you a good overview of what would be some good properties and security and privacy and how bitcoin does not have some of those properties. And, Ethereum has had a few attacks on it; so, this is the DAO attack. So, it is a decentralized application organization. So, this is a there is a blog on that, so it gives you a good overview of what that attack was really about. This was really not an attack on Ethereum, it was attack on a smart contract which is which was the DAO. DAO is really a really a smart contract on Ethereum.

So, the attack on that and how they were able to steal certain Ethereum coins from that right, ether from that from those accounts and eventually Ethereum had to do a hard fork to undo that hack right. So, some details about Intel SGX; so Intel itself has published

like it is not a very long one. So, a few pages long, it is a HTML you can go read that; it gives you a good overview of the some of the nice properties that Intel SGX provides in terms of a isolation and privacy. And finally, the Coco framework white paper itself is again a good read, it gives you some of the properties that Coco framework gives you.

I am sure this is for Coco framework, if you read it you will see it is it is really work in progress. I am sure Microsoft has more things to more things in store that they are planning to bring out along with this. So, with that that ends this lecture; over the next few lectures we will go at length. Add some of the security and privacy properties, we need for enterprise applications and we look at some examples of how that is provided by hyperledger fabric.

With that thanks a lot, we will see you at the next lecture.