OneNet Blockchain Consulting

www.onenetblockchain.com

BLOCKCHAIN

We have around 5 years of experience in exploring Blockchain and have created many smart contracts in Bitcoin, Ethereum, Hyperledger, Tendermint, Ethermint and R3 corda.

Currently we are researching on different product possibilities based on popular blockchain platforms.

Bitcoin

Bitcoin uses a Forth-like, stack-based scripting language to create smart contract and the script is processed from left to right. It is purposefully not Turing-complete, with no loops so to limit the attacks from malicious programmers. And we have developed smart contracts in bitcoin only for test purposes.

Ethereum

Ethereum has a fully-fledged Turing-complete programming language to write smart contracts for the blockchain. The smart contracts live on the blockchain in an Ethereum-specific binary format called EVM bytecode and are run by Ethereum Virtual Machine (EVM) just like JVM runs java bytecodes. The smart contracts are typically written in some high level language such as solidity, serpent and LLL and then compiled into bytecode to be uploaded on the blockchain. Ethereum also comes with the browser based IDE called Remix. Smart contracts can be written and deployed using Remix-Solidity IDE, Truffle and Mist Ethereum wallet. We have created many blockchain smart contracts in Remix-Solidity IDE and Truffle.

Hyperledger

The base platform for the hyperledger blockchain is called the fabric and it allows you to create blockchains, smart contracts on top of this fabric. Some use cases includes: Business contracts, Asset depository, Supply chain, etc. The fabric has a modular architecture allowing pluggable implementations of various functions. It features powerful container technology (Hyperledger context layer) to host any mainstream language for smart contracts development. We have developed Smart Contracts in Hyperledger fabric and sawtooth.

MAILCOIN

Technology: ***Tendermint, Ethermint, Solidity, react.js, Golang.***
Team Size: 4

We have developed an application for United States postal services to track packages on blockchain and issue postal stamps on blockchain.

MailCoin Exchange: In this model, the foundation can sell any number of tokens to verified users through a crowdsale contract with buy + sell prices at fixed highs and lows. When the master account receives Ether, it begins to hold a balance and when. The users can sell MailCoin to the foundation also if it holds a balance.

MailCoin Store / Supply Chain Workflow: In this model, after the user has acquired tokens, then can purchase goods and services from the MailCoin store. For companies, maybe x MAIL is charged to be

able to place items. Settings inputted by company become stored in a smart contract.

POS Cryptoeconomics:
Hard Coded Token Limit In a hard coded token limit model, you identify the number of tokens that will exist as a parameter written into the token smart contract. This simplifies the development and blockchain build workflow but has no built in incentive for nodes to secure the network.

Basic Attention Token: 1B BAT tokens for sale, 1.5B total supply, secured on Ethereum blocks, token represents the API key to the platform o Augur: 11,000,000 REP tokens will ever exist, secured on Ethereum blocks, token represents a voting share on platform, Qtum: 59,000,000 circulating supply, 100,000,000 total supply, tokens used as gas to power dapps built on qtum blockchain, 750ms-3s blocks, ~1% annual Proof of Stake (PoS) o Waves: 100,000,000 Waves total supply, tokens used as payment to create tokens and deploy contracts on blockchain, 1-30 sec blocks o NEM: Fixed 8,999,999,999 in Genesis block, one minute blocks.

Inflationary POS: If we want to reward nodes for securing the network, we need to build a consensus algorithm into the application. Must have economic incentives to discourage centralized groups from acting badly, and anti-centralization incentives to discourage groups from forming.

POS: All nodes can validate, block reward split evenly for stakers o DPOS: "Delegated POS" where only a number of "delegates" have the power to validate blockchain transactions, Masternodes: Dash and Pivx are examples of this, "masternodes" or nodes that have completed an action such as holding a number of tokens (1,000 with Dash, 10,000 with Pivx) receive 45% of the mining reward from each block, 10% goes to a development fund, 45% to the miner who found the block, Servicenodes: Includes an individual node rating system similar to Uber, nodes need to be vetted for security, meet hardware

Anami Blockchain

Technology: ***Hyperledger, Golang, Postgresql, react.js.***
Team Size: 10

We have developed a blockchain application for United States government to make cannabis trading more secure and transparent on blockchain. Most banks are reluctant to provide bank accounts to dealers and retailers of cannabis. So, they mostly have to deal in cash and government finds it difficult to track those transactions and hence faces loss in tax collection for the government. We have developed this solution using hyperledger.

Nexus Blockchain API (Ethereum & IPFS)

Technology: ***Ethereum, Solidity, react.js, Golang.***
Team Size: 4

The Ethereum blockchain provides a programming language called Solidity that supports the creation and execution of “smart contract” code. The Ethereum Virtual Machine (EVM) is the code execution environment for these smart contracts. For Nexus, one or more smart contracts will be implemented that perform the basic operation of recording metadata about media files and data files stored on IPFS/Swarm and serving as the proof of existence and ownership of the specified media files.
The Nexus blockchain API will consist of several components, in addition to the smart contract code. The backend API will communicate with the blockchain API through RPC (remote procedure calls) and will provide the files and data to save. Here is an example:

- Nexus backend API receives request from mobile app to store and record media file(s)
- Nexus blockchain API runs as an RPC service (on the backend servers)
- RPC request received from backend API to store specific files with metadata about ownership,

date/time, location, etc.
- Each media file is uploaded to IPFS/Swarm and permanent URL is returned.
- Metadata, along with IPFS URLs, are converted to structured JSON format and stored in IPFS/Swarm and permanent URL is returned. The JSON file format will conform to a versioned specification that will evolve over time.
- Hash values are calculated for each file and URL to represent the original and permanent record of the files and metadata
- Metadata URL and hash values are sent to Nexus smart contract to record in Ethereum blockchain and block transaction information is returned and captured.
- All metadata and transaction information is saved to backend API using web service calls from blockchain API.
- Backend API notifies mobile app that files and data have been stored successfully.

MEDICAL PRACTITIONER VERIFICATION FOR INSURANCE COMPANIES (BLOCKCHAIN)

Technology: ***Hyperledger, GoLang, express.js, angular.js, HTML5.***
Team Size: 4

Entities involved & brief description of the project:

- Ease the verification process of Medical Practitioners / Hospitals
- Cut cost on verification process of Medical Practitioners / Hospitals.
- **Verifying Agency**: The Verifying Agency will collect information about Medical Practitioner and Hospitals and upload it to blockchain through a web interface.
- **Insurance Company**: Insurance Company will request information from blockchain through a web interface by paying a small amount. The payment will directly go to the Agency who verified the practitioner / Hospital.
- **Medical Practitioner / Hospital**: The information about them will be available on the blockchain uploaded by an Authorized Agency.

AXIS / ICICI BANK PEER TO PEER MONEY LENDING (BLOCKCHAIN)

Technology: ***Hyperledger, GoLang***
Team Size: 6

Entities involved & brief description of the project:

- Lender: All NBFC license holders, who will lend money to the Vendor
- Vendor: Who needs an immediate payment for the product/service they offered to a company or organization
- Company/Organization: That received the product/service from the vendor. They issued the Delivery note, however payment is due to vendor.
- Once the Vendor uploads the delivery note and Invoice to the blockchain to seek a loan, various Lenders can compete and provide the best interest.
- The vendor can accept a loan and hence can get the payment immediately from the Lender.
- Once the company/organization releases the payment, it will go to the vendors account from where the smart contract will automatically deduct the Lenders amount (Principal + Interest) and will transfer it to the Lender, the balance will automatically get credited to Vendor's bank account.

AIRBUS AVIATION PARTS MANAGEMENT (BLOCKCHAIN)

Technology: ***Hyperledger, GoLang***
Team Size: 6

Problem: In aviation, authorities defined a complex process how to follow spare parts. Officially a

manufacturer is asked to issue a document for each spare part, only sell these to companies listed as allowed on what is called a CAP (capability) list, and check serial numbers of destination planes for these spare parts. But as these parts a very expensive, parts are also gained from end of life planes, refurbished, and brought into use again. The underlying process – defined as being secure – leads to enormous amounts of paper and process costs. But in fact, fake parts have been found in normal machines as well as Air force One or strategic bombers.
I would argue that lots of this process is simply unrealistic. A manufacturer needs to accept that companies like Lufthansa Technik store parts in depots to have them in fast access when needed. It would simply take too long to order when parts are broken. But therefore LH Tech cannot provide the final destination airplane when ordering the part.

Solution: We have a situation where various players such as vendors, airplane manufacturers, various authorities, maybe even you as a passenger have an interest to understand where parts came from, what history they have, and nobody really should need to trust another. It feels like *blockchain* could solve this. My idea would be to build a separate blockchain. Each stakeholder gets a vault as well as storage facilities, airplanes, licensed technicians etc. A manufacturer would then generate a new part by making a transaction including its serial number, the issuer (the manufacturer) and the receiver (the airplane or the storage facility). When the part would be taken out of store, another transaction takes place including the technicians, the issuer and the receiver aircraft. Things like taking out a whole kitchen and splitting it into spare parts could be recorded. And so on…
In fact, by using the machine's vault, you could generate a complete history of its parts. By using a part, you could generate a complete lifetime history. For the whole industry, changing to this system would help to reduce process costs and increase traceability.

FLEXI-LOCK (BLOCKCHAIN)

Technology: ***Hyperledger, GoLang***
Team Size: 4

In a sharing economy, there is a need for people to be able to trade their items and accept payments. Owners have (one or more) assets to rent – assets can be physical or digital in nature. At a broad level we have an asset (example locker/hotel room/bike/movie) belonging to an owner (individual's/hotel chain/media outlet) access to which is controlled by a lock (a lock that has computing power, enabled with a digital pin which is accessible via internet) on the payment of a suitable rent for the duration of the use of the asset. The pin to the locker can be shared with a bona fide customer (user) as per business rules and reset via an API as soon as the rent is exhausted.

Let us introduce FlexiLock, which performs the following functions:

- Asset Discovery: Lists the inventory of assets available with their states (empty/reserved) so that users can reserve assets they want to use by making payments, the lock details are shared post payment
- Provides APIs to access and manage the lock
  - Validate PIN
  - Open Lock
  - Reset Lock PINs for user as well as owner
- Provides APIs to manage transactions (note that we are allowing the use of fiat currency along with cryptographic currencies)
  - Send Payment
  - Manage Wallet
  - Transfer money to bank account or a Bitcoin exchange
- Owners are able to list the assets they want to rent out
- Notification APIs to intimate state change for the locks:
  - Inform the owner that the lock use has completed and the asset is available for re-listing

    - Inform the user about the PIN, lock details etc. over a secure channel.
- APIs for FlexiLock to interact with other systems like Expedia hotel booking or Amazon/FedEx Logistics. This involves transacting with these systems to pass on the lock and PIN details via them to the end user.

FACTOM BLOCKCHAIN (FCT)

Technology: ***Blockchain***
Team Size: 3

We worked with Factom blockchain to provide active solutions for compliance, identity, transparent assets, and securities.

Factom is a system for securing millions of real-time records in the blockchain with a single hash. This gives you the tools to build applications with all of the security of the blockchain without the speed, cost, or size limitations.

Factom allows you to build applications on top of the Bitcoin blockchain. Factom uses a simple API that lets you build projects that were not possible before while still harnessing the trust and security of the Blockchain.

Factom is a solution to:

- Blockchain bloat
- Off chain transactions
- Use cases and project ideas for hashing data into the blockchain

How Factom works?

- Application Owner purchases Entry Credits with Factoid
- Application records an Entry
- Factom Servers create Entry Blocks and Directory Blocks
- Factom secures an anchor (hash of the Directory Block) onto the blockchain

We were running Factom nodes in our servers and also researching on Factom to team up on further possibilities.

RIPPLE BLOCKCHAIN (XRP)

Technology: ***Blockchain***
Team Size: 3

We worked with Ripple blockchain to provide global financial settlement solutions to enable the world to exchange value like it already exchanges information.

Ripple is a real-time gross settlement system (RTGS), currency exchange and remittance network by Ripple. Also called the Ripple Transaction Protocol (RTXP) or Ripple protocol, it is built upon a distributed open source Internet protocol, consensus ledger and native currency called XRP (ripples).

We are in discussion with the ripple team and in the process to implement RTGS solution to provide instant remittance to customers who want to transfer money from abroad to India and vice versa.