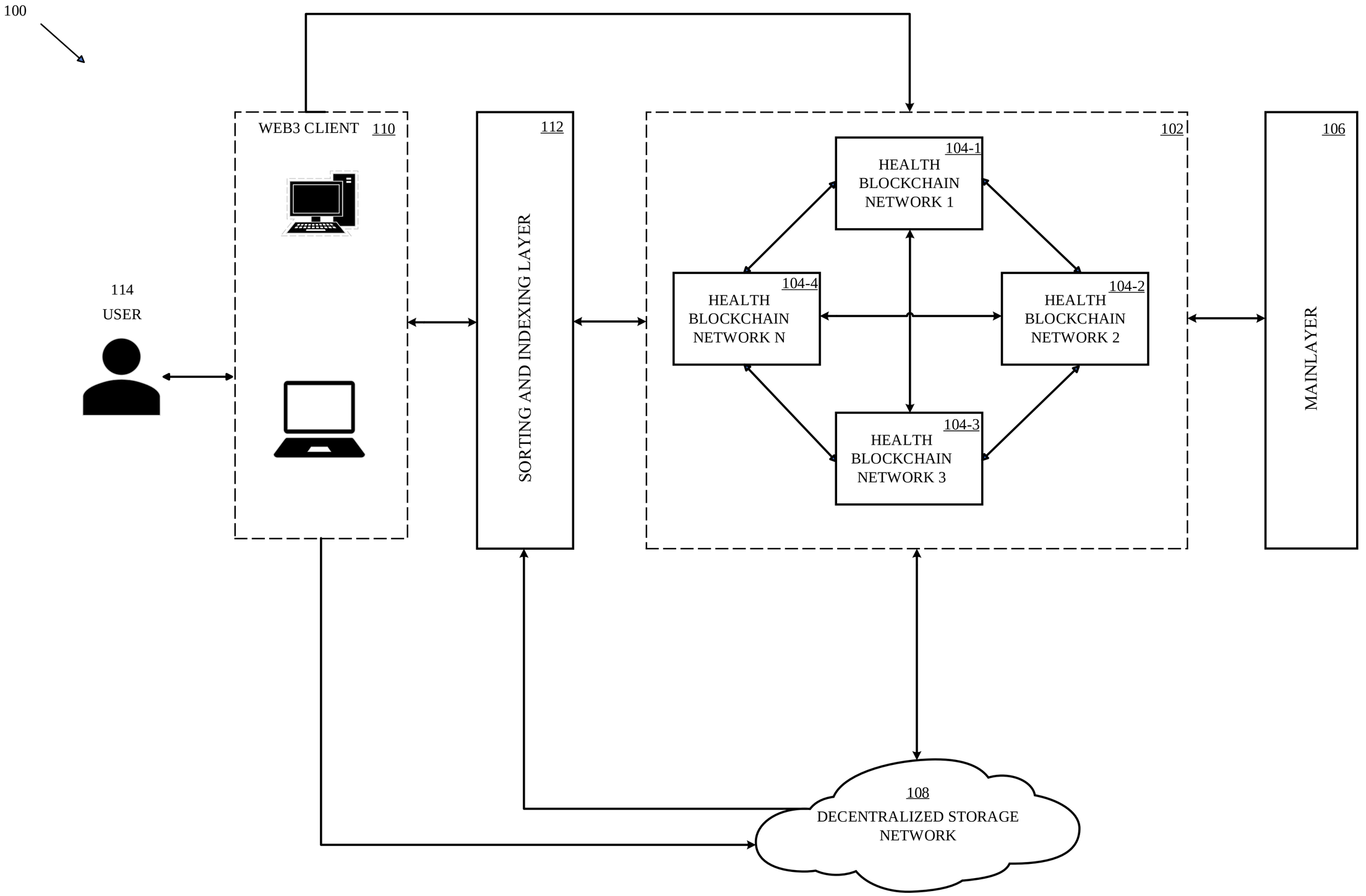


FIG. 1

Tarun Khurana
Regd. Patent Agent [IN/PA-1325]
Dated: 19th September, 2022

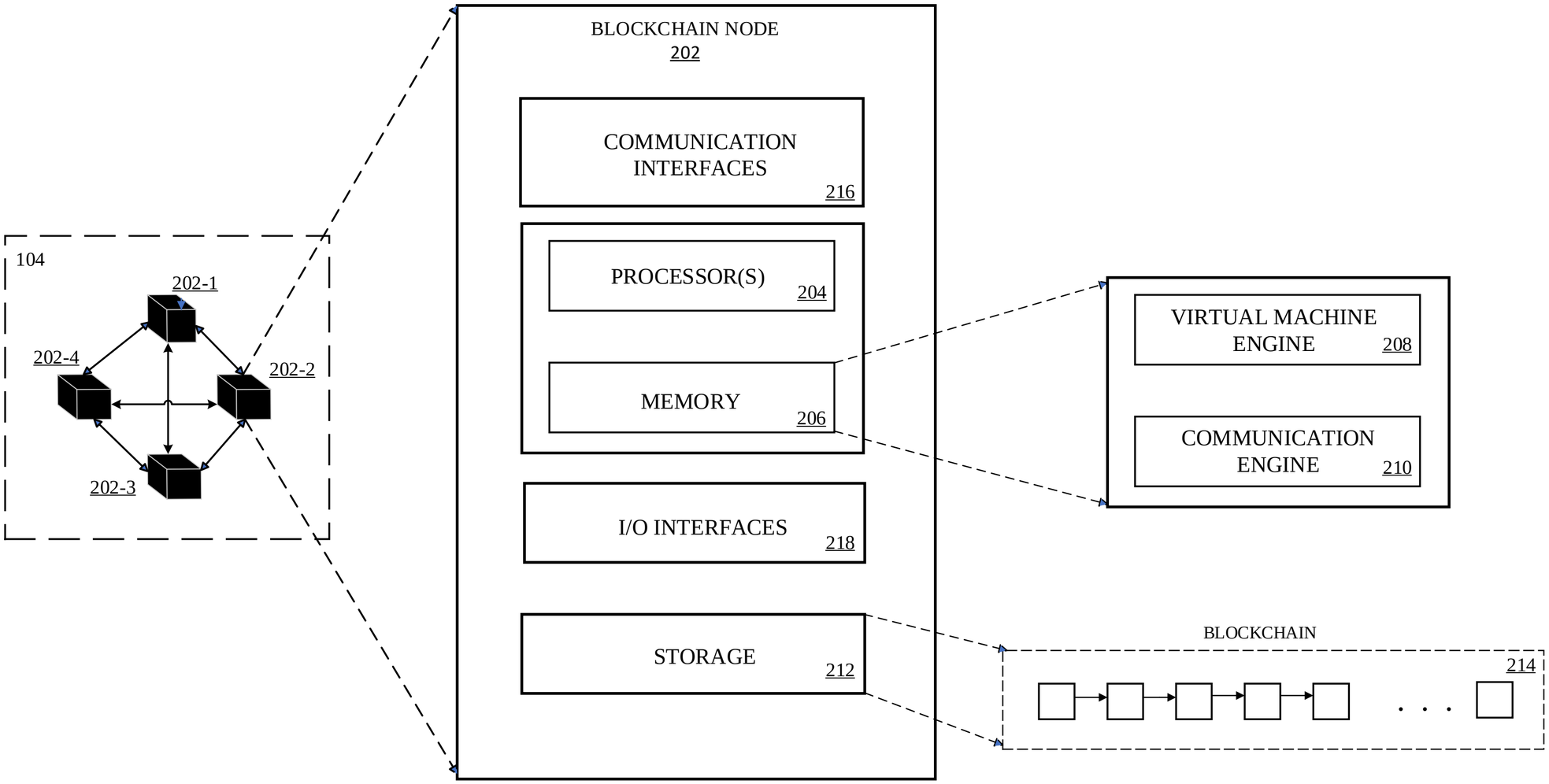


FIG. 2

Tarun Khurana
Regd. Patent Agent [IN/PA-1325]
Dated: 19th September, 2022

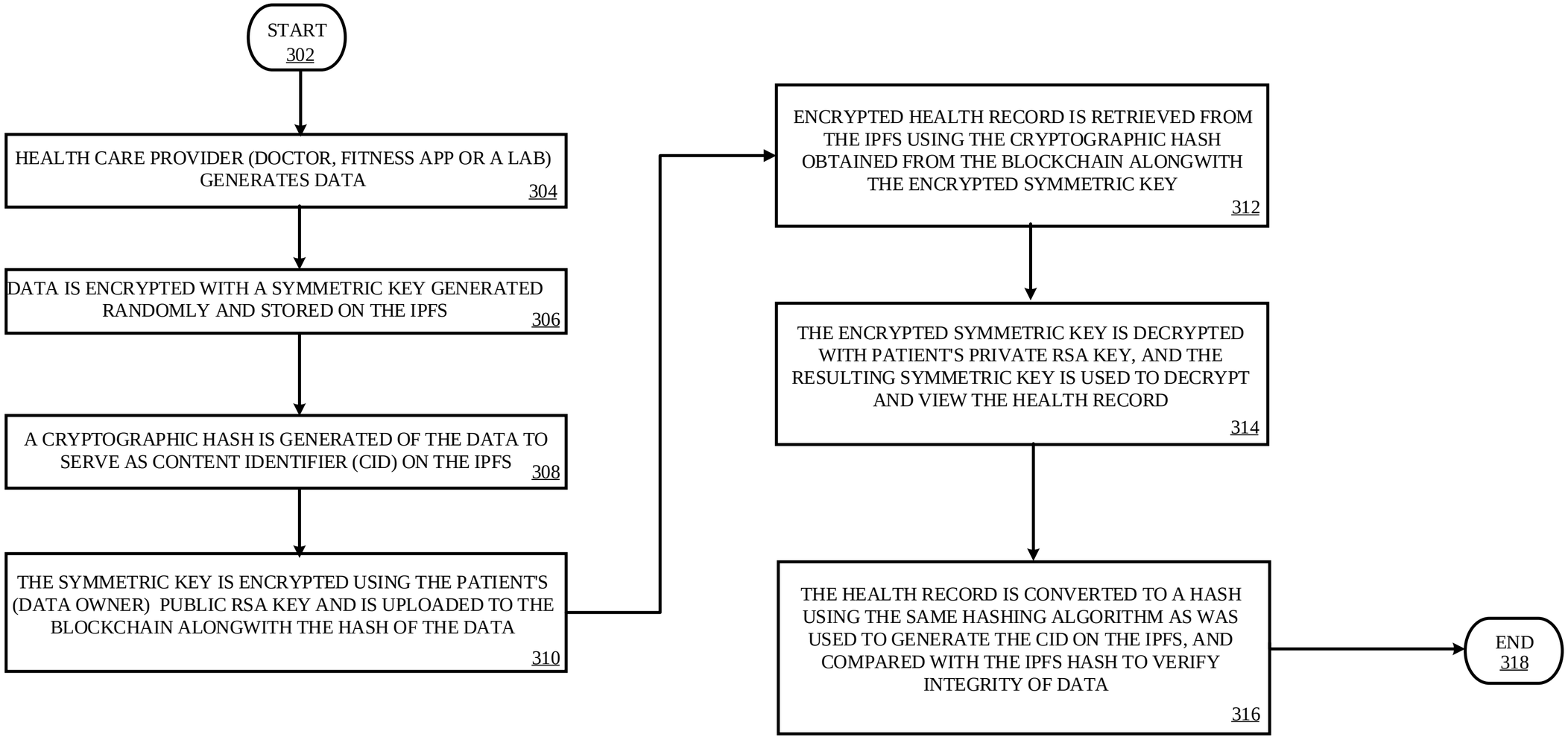


FIG. 3

Tarun Khurana
Regd. Patent Agent [IN/PA-1325]
Dated: 19th September, 2022

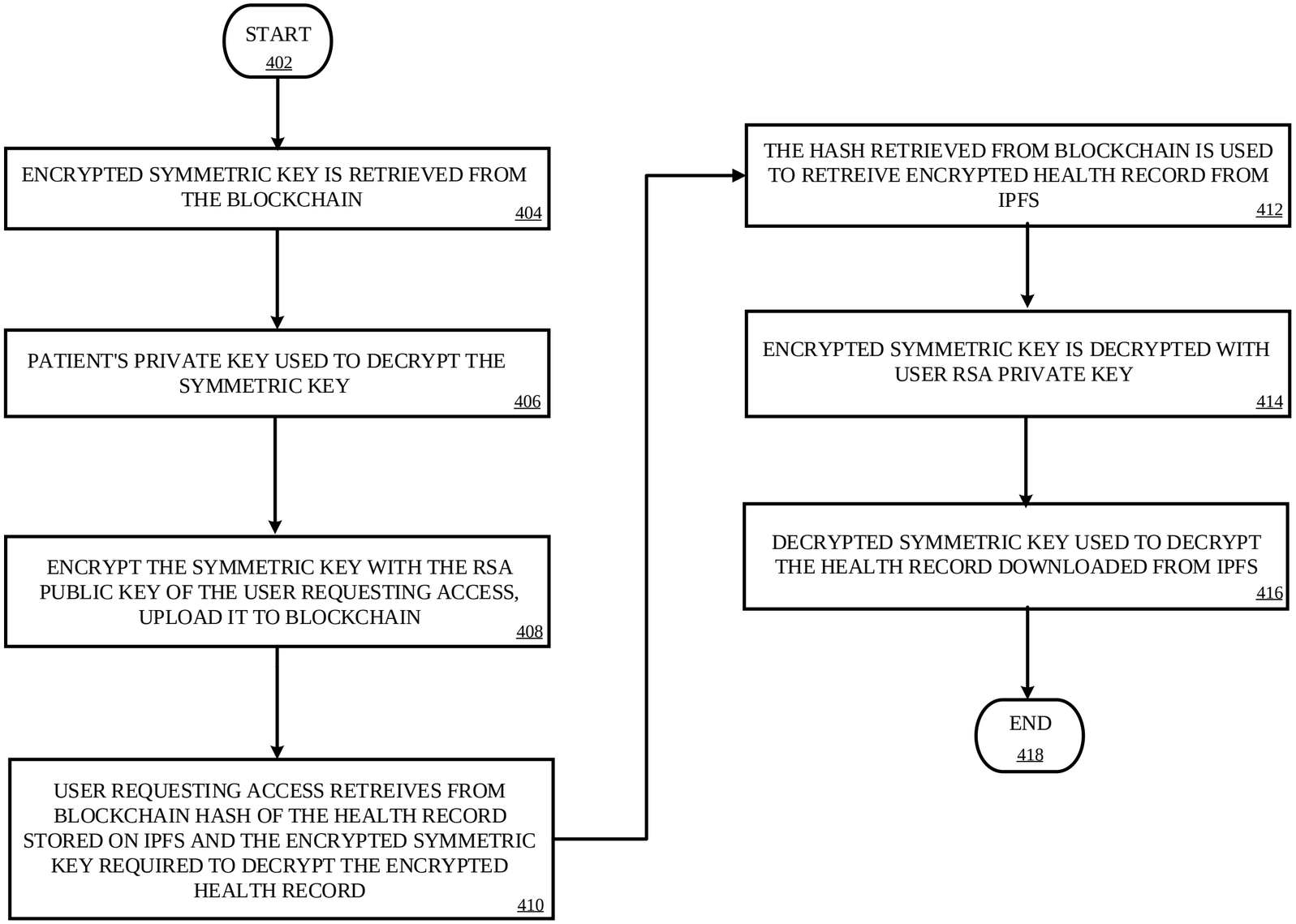


FIG.4

Tarun Khurana
Regd. Patent Agent [IN/PA-1325]
Dated: 19th September, 2022

FORM 2

THE PATENTS ACT, 1970

(39 of 1970)

&

The Patent Rules, 2003

COMPLETE SPECIFICATION

(See sections 10 & rule 13)

1. TITLE OF THE INVENTION

SYSTEM AND METHOD FOR DECENTRALIZED AUTONOMOUS HEALTHCARE DOCUMENTATION

2. APPLICANT (S)

NAME	NATIONALITY	ADDRESS
Chitkara University	IN	Chitkara University, Chandigarh-Patiala National Highway, Village Jhansla, Rajpura, Punjab - 140401, India.
Bluest Mettle Solutions Private Limited	IN	ODC-4, Panchshil Tech Park, inside Courtyard by Marriott premises, Hinjewadi Phase - 1, Pune - 411057, Maharashtra, India.

3. PREAMBLE TO THE DESCRIPTION

COMPLETE SPECIFICATION

The following specification particularly describes the invention and the manner in which it is to be performed.

TECHNICAL FIELD

[0001] The present disclosure generally relates to healthcare documentation systems, and specifically to the usage of self-executing agreements referred to as smart contracts in healthcare documentation systems, to assign rights to store and access healthcare records on a decentralized storage network.

BACKGROUND

[0002] The management of electronic health records is characterized by numerous data silos distributed across organizations (e.g., hospitals and laboratories). In such a distributed scenario, recognising and validating the credentials of parties originating or exchanging medical data becomes an issue. In such distributed data communities, interoperability is an issue. As an example, a hospital, a small physicians' practice group and a pharmacy may each operate separate electronic medical record systems for their respective uses, without these individual systems having the capability of talking to each other. In such a situation, discovering the origin and the full chain of custody of medical data spread across different silo communities becomes an issue.

[0003] The present data silos are not patient centric. They focus rather exclusively on health care providers rather than meaningfully including patients. Further, data managed in these various data silos is not tamper proof and doesn't maintain records of read/write access. Unapproved parties may be able to view the patient's clinical data during the data transfer. Thus, data integrity, security and patient privacy are compromised. There is a general negative belief on how personal data in managed.

[0004] Access control, wherein a patient decides what and how much of his medical history or condition he can reveal and to whom, is absent in present systems.

[0005] Thus, current healthcare enterprise applications need greater flexibility and scalability to meet the challenges of heterogeneity of healthcare systems at all levels—data, process, services, and payments.

OBJECTS OF THE PRESENT DISCLOSURE

[0006] It is the object of the present disclosure to build a decentralized healthcare-document management system that is reliable, preserves data sovereignty, integrity and security.

[0007] It is the object of the present disclosure, to propose a healthcare document management system that uses sidechains and the Proof of Stake (PoS) consensus scheme to allow for high throughput and scalability.

[0008] It is the object of the present disclosure to integrate fully sovereign Ethereum-compatible healthcare based blockchain networks.

[0009] It is the object of the present disclosure to use smart contracts as a data access control mechanism due to their ability to expose the minimum amount of data necessary.

SUMMARY

[0010] The present disclosure provides for a system and method to integrate health data silos by implementing them as network of sidechains to a permissionless blockchain network like Ethereum also referred to as main layer, every element of the system coupled to each other through communication channels using a peer-to-peer networking protocol.

[0011] In an embodiment, the healthcare data may be stored in a decentralized storage network, with cryptographic hashes of the data stored being used as content identifiers.

[0012] In an embodiment, the healthcare data on the decentralized storage network may be encrypted using symmetric cryptography.

[0013] In an embodiment, smart contracts may be used to control access to data stored on the decentralized storage network, the transaction being recorded on the main layer for finality. Most transaction processing takes place off-chain in the network of sidechains, with only the results being committed to the main layer.

[0014] In an embodiment, a web3 client may be used along with a decentralized indexing and querying tool to efficiently query the system.

BRIEF DESCRIPTION OF DRAWINGS

[0015] FIG. 1 illustrates a schematic diagram of an embodiment of the proposed system of healthcare documentation, in accordance with an embodiment of the present disclosure.

[0016] FIG. 2 illustrates an embodiment of a blockchain node, in accordance with an embodiment of the present disclosure.

[0017] FIG. 3 illustrates a flow diagram of an embodiment describing the sequence of steps that are executed to upload health documents to the health network by a health provider, in accordance with an embodiment of the present disclosure.

[0018] FIG. 4 illustrates a flow diagram of an embodiment describing the sequence of steps that are executed to grant access to an authorized user to view relevant health records of a patient, in accordance with an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0019] In the following description, for the purposes of explanation, various specific details are set forth in order to provide a thorough understanding of embodiments of the present disclosure. It will be apparent, however, that embodiments of the present disclosure may be practiced without these specific details. Several features described hereafter can each be used independently of one another or with any combination of other features. An individual feature may not address all of the problems discussed above or might address only some of the problems discussed above. Some of the problems discussed above might not be fully addressed by any of the features described herein.

[0020] The ensuing description provides exemplary embodiments only, and is not intended to limit the scope, applicability, or configuration of the disclosure. Rather, the ensuing description of the exemplary embodiments will provide those skilled in the art with an enabling description for implementing an exemplary embodiment. It should be understood that various changes may be made in the

function and arrangement of elements without departing from the spirit and scope of the invention as set forth.

[0021] In the present disclosure, we use the term ‘blockchain’ to include all forms of electronic, computer-based, distributed ledgers. These include consensus-based blockchain and transaction-chain technologies, permissioned and un-permissioned ledgers, shared ledgers and variations thereof. The most widely known application of blockchain technology is Ethereum, although other blockchain implementations have been proposed and developed. While Ethereum may be referred to herein for the purpose of convenience and illustration, it should be noted that the invention is not limited to use with the Ethereum blockchain and alternative blockchain implementations and protocols fall within the scope of the present invention. The term “user” may refer herein to a human or a processor-based resource.

[0022] A blockchain is a peer-to-peer, electronic ledger which is implemented as a computer-based decentralised, distributed system made up of blocks which in turn are made up of transactions. Each block contains a hash of the previous block so that blocks become chained together to create a permanent, unalterable record of all transactions which have been written to the blockchain since its inception. All smart contracts and state changes on the Ethereum blockchain are executed by transactions.

[0023] A blockchain network is a decentralized distributed ledger network, meaning that a distributed ledger is maintained on multiple nodes of the blockchain network. The blockchain network can include one or more nodes in which one or more servers and/or client devices may function as one or more nodes. Each node in the blockchain network may have a complete or partial copy of the entire ledger or set of transactions and/or blocks on the blockchain. Transactions are initiated at a node of a blockchain network and communicated to the various nodes of the blockchain. Any of the nodes can validate a transaction, add the transaction to its copy of the blockchain, and/or broadcast the transaction, its validation (in the form of a block) and/or other data to other nodes.

[0024] The Ethereum Virtual Machine (206) is the runtime environment for smart contracts in Ethereum. The EVM handles all the transaction processing on the Ethereum network. As with any virtual machine, the EVM creates a level of abstraction between the executing code and the executing machine (an Ethereum node). Currently the EVM is running on thousands of nodes distributed across the world.

[0025] Smart contracts are the executable programs that run on the Ethereum blockchain or on Ethereum compatible blockchains. Smart contracts are a type of Ethereum account. This means they have a balance, and they can send transactions over the network. However, they are not controlled by a user, instead they are deployed to the network and run as programmed. User accounts can then interact with a smart contract by submitting transactions that execute a function defined on the smart contract. Smart contracts can define rules, like a regular contract, and automatically enforce them via the code. Smart contracts cannot be deleted by default, and interactions with them are irreversible.

[0026] Sidechains are separate blockchains that run independent of Ethereum and are connected to Ethereum by a two-way bridge. Some of the main barriers to the widespread adoption of blockchain applications are slow block confirmation times, high transaction costs, limited scalability, and subpar user experiences. Because sidechains are EVM compatible, they are considered a useful scaling solution for Ethereum. EVM-compatible sidechains support smart contracts written in Solidity, as well as other EVM smart contract languages, which means smart contracts written for Ethereum, will also work on EVM-compatible sidechains. Each sidechain is responsible for its security and doesn't inherit Ethereum's security properties. Some examples of consensus algorithms used on sidechains include Proof of Authority (PoA), Proof of Stake (PoS) and Byzantine fault tolerance. Like Ethereum, sidechains have validating nodes also referred to as validators, that verify and process transactions, produce blocks, and store the blockchain state. Validators are also responsible for maintaining consensus across the network and securing it against malicious attacks. Sidechains sacrifice some measure of decentralization or security to achieve high throughput.

[0027] Proof of Stake (PoS) is a consensus algorithm for public blockchains that depend on a validator’s economic stake in the network. In a PoS based blockchain, a set of validators take turns proposing and voting on the next block. The weight of each validator's vote depends on the size of its deposit or its stake. Significant advantages of PoS include security, reduced risk of centralization, and energy efficiency. The blockchain keeps track of a set of validators, and anyone who holds the blockchain's base cryptocurrency (like ether on Ethereum) can become a validator by sending a special type of transaction that locks up their ether into a deposit. The process of creating and agreeing to new blocks is then done through a consensus algorithm that all current validators can participate in.

[0028] In a traditional electronic medical record management system taking a medical institution as a centre, medical data of a patient is managed by the medical institution, and the patient cannot directly browse and control the medical data of the patient. Furthermore, patient medical records are stored in separate medical institutions, and sharing of medical data becomes difficult, which makes interoperability between medical institutions low, and also makes personal medical information of patients stored in fragmented form, failing to construct complete lifelong medical information for patients. Then, the traditional centralized storage of medical data brings huge risks, and the events such as data loss and hacking attack are endless. In addition, in extreme cases, when medical disputes occur, data managers in medical institutions may even directly modify medical data, which is not conducive to maintaining the interests of patients. In addition, when a third party non-medical institution, such as an insurance company, makes a claim, the patient's medical information will be collated, which violates the patient's privacy.

[0029] Medical data is a collection of patient privacy, such as a patient's personally identifiable information, medical records, medical images, expert diagnosis, genetic genes, etc., which are extremely private and need to be securely managed. However, both medical institutions and third-party institutions, such as insurance companies, require frequent access to a patient's personal medical records. Balancing secure storage, privacy protection, and access control of medical

data is therefore a hot issue, and management of medical data presents new challenges to medical institutions.

[0030] The present disclosure presents a secure and distributed healthcare document sharing system that integrates several healthcare blockchain networks into a network, wherein each healthcare blockchain network in the network is implemented as an independent EVM compatible sidechain that uses a Proof of Stake (PoS) consensus algorithm to validate transactions and generate new blocks. The aggregate of the healthcare blockchains networks are capable of communicating with each other and the underlying Ethereum network also referred to as main layer through peer-to-peer networking protocols. The proposed system has an off-chain decentralized storage network that is a peer-to-peer network of nodes that stores encrypted health data hereinafter referred to as an IPFS. A web3 client interacts with the IPFS directly through one or more smart contracts that run on an Ethereum Virtual Machine, hereinafter referred to as EVM. The web3 client can query the system through a storing and indexing layer that maintains an index of data stored on the sidechains and the IPFS. The components of the system can be connected by any form or medium of digital data communication such as a communication network. Examples of communication networks include a local area network ("LAN"), a wide area network ("WAN"), peer-to-peer networks (having ad-hoc or static members), grid computing infrastructures, and the Internet.

[0031] FIG. 1 illustrates an embodiment of the proposed health care documentation system that includes a network of several independent, private, permissioned, dedicated and optimized instances of Ethereum compatible chains referred to as sidechains, acting like individual health data silos, each of these being interoperable, writing and fetching data from each other and also writing to Ethereum. Thus, the system forms a giant, scalable and interoperable network of blockchains that are compatible with the Ethereum virtual machine and sync with Ethereum.

[0032] In such a system, the nodes such as hospitals, insurance agencies, medical laboratories, government agencies etc. are part of the network of sidechains and take active participation by doing mining, validation of transactions etc. All

these entities maintain the growing blockchain, ensuring the liveness of the application and the availability of the data. Additionally, the various entities are coupled to the IPFS to store encrypted healthcare documents, available on demand through a Graphical User Interface (GUI) to the web3 client (110).

[0033] The system (100) can include the following entities:

a. Patient: A user possessing a unique address, public and private key pair, and owner of the health record.

b. Doctor: A user possessing a unique address, public and private key pair, and creator of the patient's record.

c. Hospital: A user (or organization) having a facility for medical treatment including doctors and diagnose kits, preparing electronic health records, encrypting, and storing it on the IPFS (108). The hospital is able to access patient's records with the consent of the patient and refer it for further treatment.

d. Agencies: A user or organization, which uses the eHealth records for statistical analysis, insurance claim, research, etc.

e. IPFS: Third-party decentralized storage network (108) to keep the encrypted record.

[0034] Whenever a patient visits the doctor of a hospital for a consultation, it is the responsibility of the hospital to prepare the patient health record, encrypt it using a randomly generated symmetric key, and store it on the IPFS (108) for seamless service. On storage, the IPFS generates a cryptographic hash from the data, this has becomes the content identifier of the data stored on the IPFS. At the same time, the patient's public key is used to encrypt the symmetric key. The hash and the encrypted symmetric key constitute part of the meta-data of the respective patient record that along with access information is committed to the main layer.

[0035] In an embodiment, the system (100) may include an execution layer, a networking layer or a network (102) of EVM compatible sidechains (104) and an Ethereum layer, also referred to as the main layer (106). The execution layer is an abstract layer and includes smart contracts that are invoked to process and validate transactions before they are included in the networks blockchain. In an

embodiment, the execution layer can include smart contracts for performing fraud detection on a distributed network for records management. In some implementations, the network may include one or more smart contracts that are configured to analyse transactions that have occurred in the network over a period of time and to flag transactions that are related to potentially fraudulent activity (or otherwise abnormal activity).

[0036] To provide such capability, an administrator may create one or more fraud detection smart contracts. Because many or all transactions in the network are validated by a rights manager, the rights manager is configured to automatically call the fraud detection smart contract to log transactions. The fraud detection smart contract listens for and stores reports from the rights manager about abnormal activity and triggers events that meet a threshold. When an event is triggered that indicates possible fraud or abnormal activity, an administrator or other user on the network is notified to prompt further investigation. The fraud detector identifies occurrences of repeated failed attempts to access one or more patients' medical records, attempts to double-fill pharmaceutical prescriptions, or suspicious doctor behaviour.

[0037] The network (102) can include the data silos implemented as sidechains (104) and compatible with the Ethereum Virtual Machine. This means that smart contracts programmed for the main layer (108), will also run on each of these Ethereum compatible sidechains. The transactions that would otherwise be processed on the main layer, will now be processed on these sidechains (104). They can however follow their own consensus algorithm to create a block. The consensus algorithm can include proof of authority, proof of stake or byzantine fault tolerance, amongst others. Since, the throughput of a blockchain for healthcare should be equal to or faster than what a patient or provider expects or needs in comparison to the current system in place, the sidechain must process transactions at a much faster and cheaper rate than Ethereum. For example, a Proof of Stake (PoS) sidechain with a few validators has faster block time with high throughput. It chooses scalability over a high degree of decentralization. However, a final state commit to Ethereum is bulletproof and passes through a final layer of checkpoints that also uses a PoS

strategy to commit blocks to the Ethereum blockchain. Thus, a dual strategy of PoS at the check pointing layer and the block producer layer achieves faster block times while ensuring a high degree of decentralization by using the checkpoints and fraud proof mechanisms.

[0038] Referring to FIG. 2, each node (202) in the network of blockchain networks (102) is a computing device and can include a processor (204) with memory (206) coupled to it, the memory (206) contains stored instructions that can be executed on the processor (204) to perform various functions of blockchain network node. The blockchain node (202) may include storage (212), communication interfaces (216) and one or more I/O interfaces (218) The various components 204, 206, 212, 216, 218 may be interconnected through at least one system bus, which may enable the transfer of data between the various modules and components that form the computing device.

[0039] The processor (204) may include single-threaded processor(s), multi-threaded processor(s) or both. The processor (204) may be configured to process instructions stored in the memory (206) or on the storage (212). For example, the processor (204) may execute instructions for the various software module(s) described herein. The processor (204) may include hardware-based processor(s) each including one or more cores. The processor (204) may include general-purpose processor(s), special purpose processor(s), or both.

[0040] The memory (204) may include one or more computer-readable storage media (CRSM). The CRSM may include one or more of an electronic storage medium, a magnetic storage medium, an optical storage medium, a magneto-optical storage medium, a quantum storage medium, a mechanical computer storage medium, and so forth. The CRSM may provide storage of computer-readable instructions describing data structures, processes, applications, programs, other modules, or other data for the operation of the computing device. In some implementations, the CRSM may include a data store that provides storage of computer-readable instructions or other information in a non-transitory format. The CRSM may be incorporated into the computing device or may be external with respect to the computing device. The CRSM may include ROM, RAM, or both.

One or more CRSM suitable for tangibly embodying computer program instructions and data may include any type of non-volatile memory, including but not limited to: semiconductor memory devices, such as erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM and DVD-ROM disks. In some examples, the processor (204) and the memory (206) may be supplemented by, or incorporated into, one or more ASICs.

[0041] In an embodiment, the memory (206) can store one or more software rules engines or programs such as a virtual engine (208) and a communication engine (210). The one or more virtual engines and communication engines of the blockchain network can be configured to run on one or more nodes, servers and client devices with data transferred between them.

[0042] The virtual engine can function as a virtual machine logic stored in memory which may be executable by the processor (204). Preferably each blockchain node (202) may include a virtual machine engine (208) and the virtual machine engine may manage and perform data transactions on the blockchain. Generally, a virtual machine engine may be run by the processor (204) of a blockchain node (202) and may maintain a distributed ledger on its memory (206) and may thus synchronise transaction data with other nodes of the blockchain network containing the distributed ledger in order to implement a blockchain based transaction processing system. In preferred embodiments, a virtual machine engine (208) may be configured to incorporate the data of one or more smart contracts into the blockchain of the supply chain system. In further embodiments, the blockchain network may include a permissioned blockchain network, and the virtual machine logic may be configured to perform a cryptographic hash function on the data of a smart contract that is incorporated into the permissioned blockchain.

[0043] Further, the memory (206) can include a communication engine (210) which is communication logic stored in memory (206) and executable by the processor (204). Preferably, the communication engine (210) may be configured to govern electronic communication between nodes and client devices. In some

embodiments, communication engine on a client device may be configured to communicate data of a smart contract between a client device and the blockchain via one or more nodes. The communication engine on a node may connect to a decentralized application (dApp) on a client device of a user and may receive input from a user, display output to the user preferably through am I/O interface. The communication engine may enable a user to manipulate data of the blockchain network such as data associated with one or more smart contracts of the blockchain network.

[0044] The storage (212) refers to any type of memory and may include any of volatile memory elements including random access memory (RAM, DRAM, SDRAM and the like), non-volatile memory elements including ROM, hard drive, tape, CDROM and combination thereof. Moreover, the storage (212) may incorporate electronic, magnetic, optical, and/or other types of storage media. In one example, the storage (212) may be located internal to a computer server such as, for example, an internal hard drive connected to the local interface in the computer server. Additionally in another embodiment, the storage (212) may be located external to the computer server such as, for example, an external hard drive connected to the I/O interfaces (218) such as SCSI or a USB interface. In a further embodiment, the storage may be connected to the computer server over a network.

[0045] In an embodiment, the participating nodes of a distributed ledger network or a blockchain network may each store a copy of the blockchain (214). A blockchain is a cryptographic chain of data blocks. Each block contains a batch of transactions of previous, parent blocks and precedes transactions of possible future, children blocks. With the exception of the original block, also called the genesis block, each subsequent block maintains an explicit cryptographic pointer to its parent block such that the chain may be naturally traversed, and the complete transactions graph reconstructed. The blockchain therefore maintains an exhaustive list of validated transactions from genesis to contemporaneous updates.

[0046] In addition, a consensus algorithm may be implemented among the participating nodes in the blockchain network, allowing each network participant to reach high certainty that their view on which transaction to keep is shared by the

rest of the network. A consensus algorithm is precisely about agreeing on a decision when there is more than one node involved and making sure that the agreement is robust to different kinds of failures or attacks. The choice of consensus algorithm will depend on the topology of the network and the degree of trust between participants.

[0047] For a block to be added to the blockchain, a miner must demonstrate a proof of work before their proposed block of transactions is accepted by the peer-to-peer network and is added to the blockchain. A blockchain protocol includes a proof of work scheme that is based on a cryptographic hash function (CHF). An example CHF includes the secure hash algorithm 256 (SHA-256). In general, the CHF receives information as input, and provides a hash value as output, the hash value being of a predetermined length. For example, SHA-256 outputs a 256-bit (32-byte, 64-character) hash value. For example, the input to the CHF can include a reference to the previous (most recent) block in the blockchain, details of the transaction(s) that are to be included in the to be created block, and a nonce value (a random number used only once). The blockchain protocol provides a threshold hash to qualify a block to be added to the blockchain. For example, the threshold hash can include a predefined number of zeros (0's) that the hash value must have at the beginning (e.g., at least the first four characters of the hash value must each be zero). The higher the number of zeros, the more time-consuming it is to arrive at a qualifying hash value.

[0048] In some implementations, validation of transactions stored to a blockchain includes verifying digital signatures associated with respective transactions. These digital signatures may be based on asymmetric encryption techniques, such as public/private key infrastructure. The data elements recorded in the blockchain, while being nearly unforgeable and unalterable, remain accessible to all participants of the blockchain through their blockchain nodes.

[0049] In an embodiment, the communication interfaces (216) enable communications between one or more nodes in the blockchain network, or between the blockchain nodes and other network-connected computing devices. The communication interfaces (216) may include one or more network interface

controllers (NICs) or other types of transceiver devices configured to send and receive communications over one or more communication networks using any network protocol. For example, the communication interfaces (216) may facilitate connection to a network and may include wireless transmitters and receivers ("transceivers") and any antennas used by the transmitting and receiving circuitry of the transceivers. The transceivers and antennas may support Wi-Fi network communications, for instance, under any version of Institute of Electrical and Electronics Engineers (IEEE) 802.11, e.g., 802.11n or 802.11ac. The communication interfaces (216) may also include wireline transceivers. The wireline transceivers may provide physical layer interfaces for any of a wide range of communication protocols, such as any type of Ethernet, data over cable service interface specification (DOCSIS), digital subscriber line (DSL), Synchronous Optical Network (SONET), or other protocol.

[0050] In an embodiment, each node of the blockchain network may communicate with other nodes in the network or with other computing devices, using one or more communication networks, such as the internet or private networks such as institutional or personal intranet, or any combination of private and public networks. The communication networks may include any type of wired or wireless network, including but not limited to a LAN, a WAN, a wireless WAN (WWANs), a wireless LAN (WLAN), a mobile communications network (e.g., third generation (3G), fourth generations (4G), and Edge), and so forth. In some implementations, the communications between computing devices may be encrypted or otherwise secured. For example, communications may employ one or more public or private cryptographic keys, ciphers, digital certificates, or other credentials supported by a security protocol, such as any version of the Secure Sockets Layer (SSL) or the Transport Layer Security (TLS) protocol.

[0051] In an embodiment, the I/O interfaces (218) enable components or modules of the computing device to control, interface with or otherwise communicate with other I/O devices. The I/O interfaces may enable information to be transferred in or out of the computing device through serial communication, parallel communication or other types of communication. For example, the I/O

interfaces (218) may comply with a version of the RS-232 standard for serial ports, or with a version of the Institute of Electrical and Electronics Engineers (IEEE) 1284 standard for parallel ports. As another example, the I/O interface(s) (218) may be configured to provide a connection over Universal Serial Bus (USB) or Ethernet. In some examples, the I/O interface(s) (218) may be configured to provide a serial connection that is compliant with a version of the IEEE 1394 standard.

[0052] The I/O devices that the computing device can interface with include keyboard, mouse, a pen, a game controller, a touch input device, an audio input device (e.g., a microphone), a gestural input device, a haptic input device, an image or video capture device (e.g., a camera), or other devices. In some examples, the I/O devices may also include one or more output devices, such as a display, light-emitting diodes (LEDs) display, an audio output device (e.g., a speaker), a printer, a haptic output device, and so forth.

[0053] The main layer (106) of the health network is a set of smart contracts implemented on Ethereum, hence can leverage Ethereum's high security. The main layer (106) can be used for things like finality, checkpointing, staking, dispute resolution and messaging between the side chains and Ethereum. Checkpoints are important as they provide finality on the Ethereum chain. The checkpoint generation occurs on a smart contract deployed on Ethereum chain. All blocks generated on the PoS side chain and submitted to the main layer, are check-pointed to the main layer in frequent intervals of time by the validators. The published state is also called a checkpoint hence the whole process is known as checkpointing. The checkpoint is a Merkle root hash of all the blocks since the time the last checkpoint was committed to the main layer. The Merkle root is then broadcasted to the staker network for their signatures. They will approve the proposed block, if it is valid, by providing their signatures. It is supposed to be validated by more than two-third validators before it is submitted on the Ethereum network.

[0054] Ethereum is essentially the ultimate abstract foundational layer: a blockchain with a built-in Turing-complete programming language, allowing anyone to write smart contracts and decentralized applications where they can create their own arbitrary rules for ownership, transaction formats and state

transition functions. The Ethereum White Paper, Vitalik Buterin, “Ethereum White Paper A Next Generation Smart Contract & Decentralized Application Platform” describes the Ethereum network.

[0055] In another embodiment, the healthcare system can use other layer-2 scaling solution like plasma solutions, zkRollups and Optimistic Rollups, layer-1 being Ethereum. The idea here is to batch transactions off-chain using a “rollup” smart contract and then periodically committing these transactions to the main chain. Layer-2 peforms the slow transaction execution off chain, with only the transaction data stored on the Ethereum blockchain. This helps scaling the Ethereum blockchain because not every transaction is needs to be executed on-chain. This also makes transactions faster and cheaper.

[0056] Healthcare transactions consist of documentation of specific episodes of healthcare services provided. Healthcare providers, payers and patients would contribute encrypted data. This could include clinical data, claims history, family history and device readings from patients. Blockchains rely on pseudo anonymity (replacing names with identifiers) and public key infrastructure (PKI), which allows the blockchain’s contents to be encrypted in a way that is prohibitively expensive to crack. Since the transactions are on the blockchain, they can never be deleted or changed, thus the blockchain maintains a complete and irrefutable record of all transactions.

[0057] Given the voluminous nature of healthcare documentation and the limited nature of a public blockchain such as Ethereum, the present disclosure proposes storing the health records on a decentralized storage network (108), where the file is encrypted to ensure patient privacy and only the metadata such as a hash of the patient health record can be recorded on the blockchain. The hash value acts as a content identifier and can be used to locate the health record saved on the decentralized storage network. In some embodiments, a smart contract may be used to store hash values of files and their corresponding signature states.

[0058] In an embodiment as illustrated in FIG. 1, the decentralized storage network (108) can be an Internet Platform File System (IPFS) network based on a peer-to-peer protocol. The IPFS is a protocol for storing and sharing content. The

IPFS is decentralized because it loads the content from thousands of peers instead of one centralized server. Each peer, also called a node, stores only a portion of the data, each file being divided into a plurality of file blocks, distributed across the one or more nodes of the IPFS network, making the files tamper proof. It should be understood that although embodiments of the present disclosure have been described primarily with the IPFS distributed network as an example, other distributed networks that locate files based on hash values also exist.

[0059] The IPFS runs on three basic principles i.e. unique identification via content addressing (CID), content linking via directed acyclic graphs (DAGs) and content discovery via distributed hash tables (DHTs). These three principles build upon each other to enable the IPFS ecosystem.

[0060] Every piece of data on the IPFS is cryptographically hashed, resulting in a safe, unique content identifier (CID). The cryptographic hash is unique to the content that it came from and is used to locate the file on the IPFS network. Cryptographic hashes are functions that take some arbitrary input and return a fixed-length value. The particular value depends on the given hash algorithm in use, such as SHA-1, SHA-256, or BLAKE2. A given hash algorithm always returns the same value for a given input. SHA-1 creates a 160-bit hash, while SHA-256 creates a 256-bit hash. Hashes can be represented in different bases such as base2, base16, base32 etc. IPFS uses that as part of its content identifiers and supports multiple base representations at the same time. Cryptographic hashes have the important characteristics of being deterministic, uncorrelated, unique and one-way. These features also mean we can use a cryptographic hash to identify any piece of data. A hash is a fixed length, so the SHA-256 hash of a one-gigabyte file is still only 32 bytes. Hence sending it around the network doesn’t take up a lot of resource. This is critical in system like the IPFS, where data is stored and retrieved from many places. A computer running IPFS can ask all the peers it's connected to whether they have a file with a particular hash and, if one of them does, they send back the whole file. Without a short, unique identifier like a cryptographic hash, content addressing wouldn't be possible.

[0061] IPFS links CIDs together using a Merkle Directed Acyclic Graphs (DAGs). The DAG is a data structure with each node being a hash of its contents. Every node in a DAG is a root of sub Merkle DAG and this subgraph is contained in the parent DAG. A file that is stored on the IPFS is broken down into blocks and then arranged in a tree like structure, the DAG. The hash of the root node in the DAG is the CID of the file. The benefit of splitting data into multiple chunks and using content addressing is data deduplication and decentralization. Data deduplication means that the IPFS will never store a duplicate block. It would instead link to an already existing DAG Node and only store the new, unique chunks. Decentralization implies that the distributed nature of data storage helps to load the content in parallel. The integrity of the received blocks can be ascertained by rehashing all the data content and asserting the constructed.

[0062] IPFS uses a distributed hash table (DHT) to find the peers hosting the content. A hash table is a database of keys to values. A distributed hash table is one where the table is split across all the peers in a distributed network. To find content, one must know determine which peers are storing the blocks that make up the content and then determine the location of these peers to request the blocks. Both these tasks require querying the DHT, hence the DHT is queried twice in order to retrieve the content stored on the IPFS in its entirety.

[0063] In an embodiment, content on the IPFS can be encrypted using symmetric cryptography, while the symmetric key used to generate the cipher can be encrypted using asymmetric cryptography, which can be added to the blockchain or a smart contract on the blockchain. This preserves the privacy of the patient and his health records. Symmetric cryptography uses the same secret key to generate a cipher and to decrypt the cipher back to the original message. Hence the secret key is shared between the sender and receiver. Advanced Encryption Standard (AES) is an example of symmetric cryptography. On the other hand, one-way functions are the main building blocks of asymmetric cryptography. A one-way function is easy to compute but computationally hard to invert. An encryption-decryption algorithm based on asymmetric cryptography consists of three steps i.e., generating a secret key also called a private key and a public key, applying the encryption method to

the file or the message that needs to be encrypted by using the private key on the file or message, to generate a cipher, and finally applying the decryption method on the cipher using the public key of an authorized user, to get back plain text. In asymmetric cryptography, each user keeps his private key and makes the public key available to everyone. This logic is used even in digital signatures. Examples of asymmetric cryptographic algorithms are RSA DSA, FFDSA, ECDSA.

[0064] Referring to FIG. 1, the web3 client (110) interacts with the decentralized healthcare documentation system (100) and can be hosted on a decentralized storage network like IPFS or Swarm. It may be configured to download and or install one or more healthcare applications that allow a user to manage, process and/or obtain health records. The web3 client UI logic can be designed using HTML, Javascript and CSS. The web3 client (110) needs to communicate with the smart contracts on the network of healthcare sidechains. The smart contracts implement the business logic of the healthcare documentation system. Every node in the network (102) keeps a copy of all states on the Ethereum state machine, including the code and data associated with every smart contract. Interacting with the data and code on a blockchain, requires interacting with one of these nodes. This is because any node can broadcast a request for a transaction to be executed on the EVM. A miner will then execute the transaction and propagate the resulting state change to the rest of the network. Hence in order to broadcast a new transaction, the web3 client needs to be either hosted on a node which runs the blockchain software or on nodes provided by third-party services (providers) like Infra, Alchemy and Quicknode.

[0065] The web3 client (110) also allows creation of a wallet to store private keys. A private key is required to sign any transaction on the blockchain. Metamask is a popular tool that makes it easy for applications to handle key management and transaction signing. Metamask is also third-party provider of nodes, hence is both a provider and a signer.

[0066] The web3 client system(s) (110) may be accessed through a suitable computing device, including, but not limited to, smartphones, telephones, tablets, smart televisions, set-top boxes, audio systems, imaging devices, computers,

computing devices, netbook computers, notebook computers, smart watches, smart appliances, wearable computers, combinations thereof, or the like.

[0067] In another embodiment, a sorting and indexing layer (112) can be deployed and can use software like the Graph to retrieve records of a patient from the blockchain. The Graph is a decentralized indexing and querying protocol that enables developers to query networks like Ethereum, Polygon sidechains and IPFS. It is a kind of data-sorting layer between the blockchain and a dApp or a web3 application. The Graph enables to build and publish open-source APIs (subgraphs) and thus, contribute to making useful data easily accessible. Data defined through subgraphs is easy to query and explore. Moreover, by transforming, organizing, and sharing this data across applications, developers can query it with a few keystrokes using an SQL like programming language called GraphQL. The Graph then makes querying this data fast, reliable, and secure.

[0068] The Graph learns what and how to index blockchain data based on subgraph descriptions, known as the subgraph manifest. The subgraph description defines the smart contracts of interest for a subgraph, the events in those contracts to pay attention to, and how to map event data to data that The Graph will store in its database. A decentralized application adds data to the blockchain through a transaction on a smart contract. The smart contract emits one or more events while processing the transaction. A Graph Node continually scans the new blocks on blockchain and the data for the subgraph they may contain and stores or updates this data in response to blockchain events. The decentralized application (dApp) queries the Graph Node for data indexed from the blockchain using GraphQL. The Graph Node fetches the data, making use of the storage's indexing capabilities. The decentralized application displays this data in a rich UI for end-users, which they use to issue new transactions on the blockchain and the cycle repeats.

[0069] FIG. 3 is an exemplary flow diagram that illustrates the steps executed when a health provider who is a registered user of the health network, wants to upload health information of a patient to the health network. The health provider can be a doctor, a pathology lab or a hospital. In step 302, the health provider generates a health document. The health provider logs on to the web3

based platform, and enters information like patient id, health provider id and uploads the document. At step 306, the uploaded file is encrypted using AES symmetric encryption. This involves generating an AES key and using the AES key to encrypt the document. The encrypted data is stored on IPFS. Once stored on the IPFS, step 308 involves generating a cryptographic hash from the uploaded health document, this hash being unique to the data used to generate it. At step 310, the AES key used to encrypt the document, is encrypted using the public key of the patient. The document hash and the encrypted AES key versions for all authorised users are stored on the blockchain. In an embodiment, this information can go into a smart contract stored on the blockchain. The blockchain then broadcasts this transaction with a transaction id that is returned to the health provider, marking completion of the process of uploading the document to the system. At step 312, the patient who is apprised of this transaction, logs in to the health system through the web3 based client (110) and queries the health system using the transaction id. The transaction id is used to look up the transaction on the blockchain, which retrieves the document hash (CID) and the encrypted AES key. The hash is used to retrieve the encrypted health document from the IPFS, while the encrypted AES key (patient version) is decrypted using the patient's private key. The decrypted AES key is further used to decrypt the encrypted document obtained from the IPFS (314). Finally at step 316, the health document is converted to a hash using the same algorithm used by the IPFS to generate the CID, and is the result is compared with document hash obtained from the blockchain to verify integrity of the document. After the comparison, the document sharing process is completed (318).

[0070] FIG. 4 is an exemplary flow diagram that illustrates the steps executed when another user like another doctor or hospital wants to gain access to the health document, the other user also having a registered id on the health system. The patient logs in to the health system through the web3 based platform and requests access for the user represented by a user id (402). This triggers a backend process that involves retrieving the encrypted symmetric from the blockchain at step 404. At step 406, the patient's private key is used to decrypt the encrypted AES symmetric key. The resulting key is again encrypted, but with the user's public

key. The information is uploaded to the blockchain, which then broadcasts this transaction to the user who is granted access to the particular record (408). At step 410, the authorised user logs in to the health system, and queries the transaction on the blockchain to retrieve the file hash and the encrypted symmetric key. At step 412, the hash also the content identifier pointing to the location of the encrypted health document on the IPFS, is used to download the encrypted content and the encrypted AES key (doctor's version) is decrypted using the doctor's private key (414). The decrypted AES key is further used to decrypt the encrypted document obtained from the IPFS (416). The health document may be converted to a hash using the same algorithm used by the IPFS to generate the CID, and the result compared with the document hash obtained from the blockchain to verify integrity of the document. This completes the process of granting authorisation to another user (418).

[0071] While this specification contains many specific implementation details, these should not be construed as limitations on the scope of any inventions or of what may be claimed, but rather as descriptions of features specific to particular implementations of particular inventions. Certain features that are described in this specification in the context of separate implementations can also be implemented in combination in a single implementation. Conversely, various features that are described in the context of a single implementation can also be implemented in multiple implementations separately or in any suitable sub combination.

[0072] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system components in the implementations described above should not be understood as requiring such separation in all implementations, and it should be understood that the described program components and systems may

generally be integrated together in a single software product or packaged into multiple software products.

ADVANTAGES OF THE INVENTION

[0073] The present disclosure helps organize and provide quick access to medical records of a patient. Storing data on a public blockchain ensures integrity and reliability of stored data and enables a user to reconstruct the full custody chain of medical documents.

[0074] The present disclosure is implemented on EVM compatible sidechains and can use other layer-2 blockchain scaling solutions to expand the performance and function of public blockchains like Ethereum.

[0075] The present disclosure provides for a system and method that integrates various data silos, thus bringing about interoperability of various healthcare data silos.

[0076] The present disclosure uses decentralized storage systems and content identifiers, thereby solving the problem of limited capacity of a public blockchain like Ethereum and ensuring tamper-resistant, distributed source of data storage.

[0077] The present disclosure leverages smart contracts to recognise, validate and assert the credentials of the parties originating or exchanging medical data across different healthcare blockchain networks.

We Claim:

1. A decentralized autonomous healthcare documentation system (100) integrating one or more independent, sovereign blockchain networks (104), the system (100) comprising:

a decentralized storage network (108) comprising one or more off-chain decentralized health-data storage devices to store patient healthcare records, the one or more storage devices being in communication with each other using a peer-to-peer networking protocol;

a network (102) of one or more independent, sovereign blockchain networks (104), that processes healthcare transactions, each of the sovereign blockchain networks being sidechains (104) pegged to a permissionless public blockchain network like Ethereum, also referred to as main layer (106), each sidechain (104) also being communicatively coupled to the decentralized storage network (108) and to each other through a peer-to-peer networking protocol, the one or more sidechains (104) comprising one or more blockchain nodes (202), wherein each blockchain node (202) comprises one or more processors (204) coupled to a memory (206) storing a set of instructions for execution by the one or more processors (204) and capable of performing steps comprising:

creating and appending new data blocks to a blockchain (214) by cryptographically linking the new data-blocks to the blockchain (214) through a process of transaction collation, local consensus and block production, wherein at-least a portion of the blockchain is stored on each of the one or more blockchain nodes (204);

registering a new user on the blockchain (214) and associating the new user with a unique user id;

receiving and validating a new transaction advertised by the registered user;

executing a virtual machine engine (208) configured to process smart contract bytecode;

executing a communication engine (210) configured to govern electronic communication amongst the one or more blockchain nodes (202) and between the one or more blockchain nodes (202) and web3 clients (110); and

storing and executing smart contracts that grant access to authorized users to write, retrieve or modify health records of a patient to the blockchain (214);

2. The system as claimed in claim 1, wherein hashes of the blocks generated by the network (102) are aggregated into a single Merkle root hash and are committed periodically to the main layer (106) for finality, the main layer (106) performing this process through execution of smart contracts on a virtual machine engine (208) like the Ethereum Virtual Machine (EVM).

3. The system as claimed in claim 1, wherein the blockchain (214) comprises a plurality of blocks with each block having a header with a cryptographic hash of a prior block and a set of healthcare transactions coupled to the header.

4. The system as claimed in claim 1, wherein the off chain decentralized storage network (108) is configured to:

encrypt patient data to maintain privacy;

shard the patient data across several nodes of the decentralized storage network (108); and

track patient data transactions through smart contracts running on the network (102);

5. The system as claimed in claim 4, wherein the one or more sidechains that constitute the network (102) store on the blockchain, a cryptographic hash, also referred to as content identifier of a health record that is stored on the

decentralized storage network (108), an asset signature and an authorization to grant access to the health record.

6. The system as claimed in claim 4, wherein the off chain decentralized storage network protocol comprises any one of Interplanetary File Storage (IPFS), Swarm, Storj, Sia, Arweave.

7. The system as claimed in claim 1, the web3 client (110) is either a node on the network (102) or has access to a node on the network (102), wherein the web3 client (110) provides access to the decentralized healthcare system (100).

8. The system as claimed in claim 1, further comprises a sorting and indexing layer (112) communicatively coupled to the network (102) and the decentralized storage network (108), the sorting and indexing layer (112) using a decentralized protocol to index blockchain data generated on the network (102) and data stored on the decentralized storage network (108) and configured to efficiently query this data, to be consumed directly into the web3 client (110).

9. A method to create a decentralized autonomous healthcare system (100) to securely store health records and resolve autonomously permissions to access these records, the method comprises:

uploading and storing an encrypted health record to the decentralized storage network (108), the health record being generated by a health provider;

generating a hash of the health record, the hash being a content identifier to locate the encrypted health record on a decentralized storage network (108);

submitting a transaction to a blockchain (214) wherein the transaction comprises the hash of the health record and permission to the

patient to decrypt and view the health record stored on the decentralized health network (108); and

assigning view permissions to the health record by the patient to any authorised user.

10. The method as claimed in method 9, wherein the health record on the decentralized storage network is encrypted with a symmetric key, the symmetric key stored on the blockchain being encrypted with a public key of the patient or any authorized user.

For Chitkara University and Bluest Mettle Solutions Private Limited



Tarun Khurana
Regd. Patent Agent [IN/PA-1325]
Dated: 19th September, 2022

ABSTRACT

SYSTEM AND METHOD FOR DECENTRALIZED AUTONOMOUS HEALTHCARE DOCUMENTATION

The present disclosure provides for a system and method to integrate health data silos by implementing them as sidechains (104) to a permissionless public blockchain referred to as main layer (106). The sidechains (104) in communication with each other and the main layer (106) through a peer-to-peer networking protocol form a network (102). The present disclosure overcomes the issues of congestion and block size restrictions of a permissionless blockchain like Ethereum by storing medical health information on a decentralized storage network (108) and controlling access to this data through smart contracts that run on the network of sidechains (102), and finally submitting all validated transactions aggregated to blocks to the underlying main layer (106), to achieve data finality.

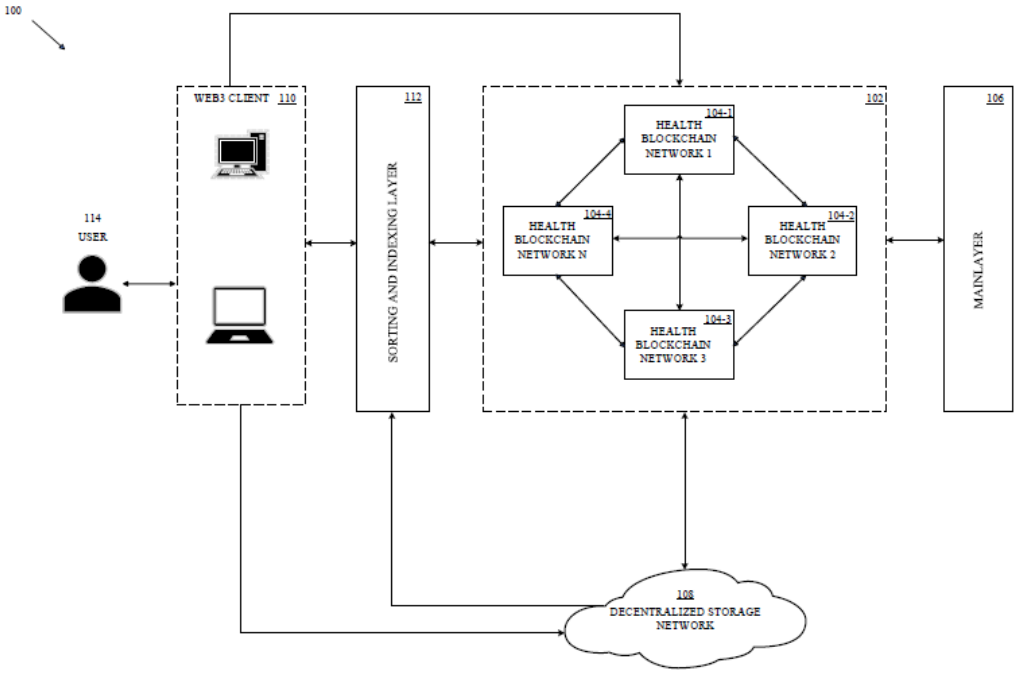


FIG. 1

For Chitkara University and Bluest Mettle Solutions Private Limited

Tarun Khurana

Regd. Patent Agent [IN/PA-1325]

Dated: 19th September, 2022