MI-CARE E-HEALTH SERVICE

Introduction.

This is a start-up project (distributed e-health system) initially to be implemented in python as a web application for health monitoring and management system (Labs, Diagnosis, medication, fitness and checkups). *Mi-care* project was initially intended for only electronic lab testing which was to be conducted on a computerized chip with software integration for machine learning but has been expanded to an E-health facility. The research and the development of the electronic testing device is still ongoing. Furthermore, the research on the algorithm (*AlgNOs*), a computerized procedure for blood sample preparation, mixing, screening/imaging and diagnosis for limited bacterial diseases like malaria is ongoing. This project aims to emulate normal testing kits with advancement on the use of imaging on top of chemical deductions.

The *Mi_care* project intends to advance the above sub project into a fully operational e-health service by integrating field operations (onsite sample extraction and testing) into a distributed e-health system for medication, record management, reporting, timely medical checkups and healthy living services. The start-up also intends to create a centralized facility for limited research on targeted diseases; more of a monitoring and control center, like the CDC; *centers for disease control*. We will invest much on the analytics of the so far collected data from various block chained hospitals to implement a real time monitoring system to offer timely updates and alerts concerning the emergency and spread of diseases across the counties.

The *Mi_care* project is to be built on an e-medicine idea but with much stress on distributed systems and block chain for proper data management, information and expertise exchange so as to eliminate medical inaccuracies and misdiagnosis (*Medical Purity*). This idea lies at the heart of universal health system.

Advantages of the *Mi_care* Project.

1) Timely checkups and clinics for early detection of diseases which in turn prompts timely disease mitigation/prevention.
2) Onsite observation, interaction and data collection which will ensures rich and accurate aggregation of integral data for tidy record keeping for future references.
3) Affordable health services through reduced cost of transport with introduction of universal health system and further implementation of onsite medication.
4) Reduced fatality and misdiagnosis through effective medication by maintaining an accurate distributed database with data integrity, information & expertise sharing across linked hospitals, [Block chain system].
5) Mass mobilization and education of citizens via onsite visits, online health tips which will prompt behavioral change and prevention of diseases.
6) Effective management of pandemic spread and mitigation through the introduction of a decentralized health system.
7) Observation of integrity, responsibility and accountability especially during tendering. This will reduce corruption among medical officers since the flow of information will be through the system and actions by stakeholders in the medical field will always be accounted for at all time.

An overview of the system Working-principles.

I. Hospitals will be registered on the system by the admins. Each hospital will be identified by a unique name, code number, location details for mapping and private key for carrying out sensitive transactions and secure data transfer and retrieval on the networked system.

II. An Admin will have regulated access to particular data on the system. This action will be implemented basing on the level of administration in the health sector.

III. A Client can only view the landing page and his account details and data recorded. Patient details could be his bio information like; Full names, physical attributes (Height, weight, skin color, Eye color, Hair color, Age, Pulse Rate, Body Temperature, Body water, Blood Pressure, Physical challenges[if any]), Genetic attributes (Gender, Blood group, fingerprints, Rhesus Factor, Race, Genetic conditions[If any],Medical Record, Contacts(Phone, Email Address, physical Address, ID), Next of Kin (Name and Phone), Family Doctor (Name and Phone), Payment Details/ Billing info (Medical cover information).

IV. A doctor may have access to the record of patients he/she have ever attended to but this may be subject to the patient's consent or by law; the reason why private key are required.

V. A doctor will have access to patient's data provided within the hospital or within the chain of hospitals. (This may need further review).

VI. A patient can request for a Doctor within the system block chain in case of any need. This will be simplified by the system. All available hospitals and/or Doctors will be visible on the patient's Dashboard and further Details may be accessed about the hospital and the Doctor (I.e. The type of medication offered in a given hospital, Doctor specialization and Medical covers available to the linked hospital).

VII. A map tool will be provisioned for easy identification of nearby registered hospitals and/or medical Practitioners

VIII. The system will also contain a Lab module for carrying out tests, diagnosis, evaluation and publishing of Lab results.

IX. The Lab will be operated onsite and at individual hospitals but the data will be available to any registered hospital

on the block chain and access will be secured through public key encryption.

X. As an envisioned distributed system, block chain technology will be the key in the implementation and management of distributed databases to enhance security and integrity of medical data.

XI. Every hospital will have their own private keys to be used as an access tool to block chained data (patient records, medical research information, Disease analytics etc.). This mode will ensure data integrity especially on tendering activities like medical equipment supply to avoid corruption scandals like the frequent KEMSA sagas.

XII. Patient will get access to any possible hospital as so long as they have registered on the block chained system. They can book appointments and get medical attention anywhere within the country. Further to this, universal medical cover will be a possibility through NHIF and other recognized medical covers.

XIII. Data collected from various hospitals can be merged for planning purposes.

XIV. Each hospital branch will have its own db. **Distributed system, Distributed Databases**

Some Special Cases here

1. There will be a special feature for any patient to register as an anonymous user.
2. He will chat with the doctor anonymously in case of confidential issues.
3. An anonymous user can access a list of doctors and select amongst them.

The AlgNos Algorithm

The AlgNos is a medical diagnosis software which borrows from digitized imaging systems, medical Diagnosis Procedures and Machine Learning for medical Diagnosis and medication. It is a trial Algorithm which is still in development phase. The AlgNos Algorithm was first envisioned by Xpotechne Team under the Mi_care program. It is the control logic and interface between the Mi_care Test kit (Lensing and Imaging) and the Networked Mi_care E-health system, it acts the same way as the Network Operating System.

The sketch or Pseudocode is as below;

i. Initiate and establish a connection link between the *Mi_care Test Kit* and the host device of the *Mi_care system* (Web application, Desktop Application and/or Android).
ii. Check for configuration of any existing microscopes and/or lenses and imaging devices on the *Mi_care Test Kit*.
iii. If there is no such configuration, report an error on the *Mi_care* patient Dashboard with possible configuration solutions.
iv. Else if the configuration is established, check for the existence of installed drivers for imaging devices.
v. If no such installation is found, run the search on the web and install all the required drivers.
vi. Initiate the imaging process and stream live images to the host machine for verification and validation.
Check and validate image quality on the patient's dashboard (Adjust Camera and Lenses Focus if necessary)
vii. With successful verification, validate and call the Machine Learning Module.
viii. Prompt user for disease symptoms.
ix. On submit perform Machine Learning operation with *Keras* library or any other image recognition algorithm.
x. Perform cross validation, generate accuracy levels and detect matches based on stored image samples.
xi. Report A positive (Match) or a negative (No Match) based on the results found.
xii. If No match, Repeat the imaging process and so forth.
xiii. Else return Diagnosis, Disease chemical make-up as an animation on the patient Dashboard and Recommendation.
xiv. Submit Result to the block chain and close imaging thread and any other running system processes on both devices.

Mi_care Test Kit Design.

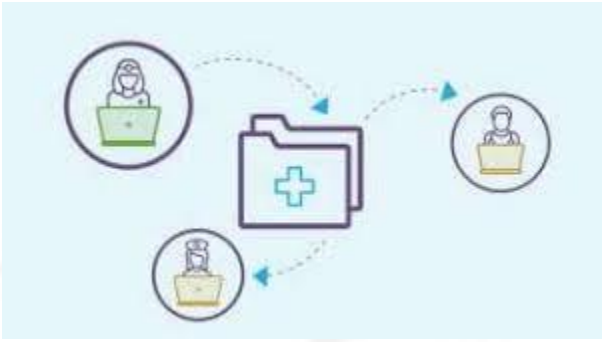
The Mi_care Test Kit is an electronic device composed of electronic sensors (lensing devices and a digital camera) and control circuit implemented on Raspberry PI for internal and external device control (The motherboard hosts the control program; the AlgNos software program) for device connectivity and networking and data transfer via Bluetooth, USB and/or WIFI. The device contain mini electronic motors for camera, lenses and sample pane motion adjustments. The ports (USB port, Pressure port), LEDS (ON/OFF LED, USB LED, Camera LED, Power LED, storage LED) and Control pad will be provisioned on the Mi_care Test Kit. Finger print module may also be included on the initial design. Biometric authentication via fingerprint will be essential and mostly recommended over the use of pin or password to unlock the device. Fingerprint Authentication identifies the owner of the device by reading his/her genetic signature unlike a simple password which can be stolen and used to hack into the device and the user account.

The Mi_care Test Kit will be used only by a single person who is the real owner of the Kit. The Mi_care Test Kit will be packaged as a medical suite alongside other medical products such as Test slides, Test Tubes and chemical reagents if needed. A test manual in form of a video will be pre_burned in the Test Kit. This will be customized according to client's needs. Health Tips and health living routine will be pre-burned in the Test Kit on request by the client. The health living instructions will be customized depending on client current health needs. The client will use the Mi_care Test Kit alongside the system to enhance proper health management.

Device Configuration.

I. Press the starter button on top of the Kit, the *ON/OFF LED* color turns to red indicating that the device has been powered on and all necessary services are installed; the main AlgNos software and Drivers.
II. On the control pad, long press the button with the Bluetooth symbol until the *Bluetooth LED* turns blue.
III. OR connect your device to your phone or Laptop using USB cable. The *USB port* is located just below the slide entrance opening.
IV. Check if the *USB LED* is powered on to green.
V. Check your computer, laptop, android phone, IPhone, or Smartwatch for any Bluetooth device; pair and connect to the *Mi_care Test Kit Device*.
VI. Launch the *Mi_care android app*, *Desktop application* or *web application*. Login and navigate to *my devices >> add a new device* and click to choose the *Mi_care Test Kit* device (Normally the device will be identified by a unique number at the bottom of the device).
VII. Provide the first time pin as indicated on the purchase warranty. Type and configure or Use the control pad and press "*Enter*" Button.
VIII. A pop message "Device configured successfully" must show on the screen. Then confirm if the *ON/OFF LED* changes from red to yellow.
IX. Provide a new pin using the control pad on the device and press "*Enter*" Button or Type in the password field as indicated on the system and press Enter.
X. Press "*Next*" Button on the application as indicated.
XI. The system asks you to place your right hand thumb finger on the device fingerprint scanner.
XII. The color of the scanner should glow from red to green. If it glows yellow, remove your finger from the scanner and wipe your finger and place your finger again on the scanner until it glows green. Do the same for the left hand thumb finger.
XIII. You should see the *ON/OFF LED* turning to green, the system should display a message "Device registration complete!!"
XIV. After the pop up fades away, you should see the logo, *AlgNos* display on the app screen for seconds and finally the user menu is presented.
XV. Consult the manufacturer's manual if anything goes wrong.

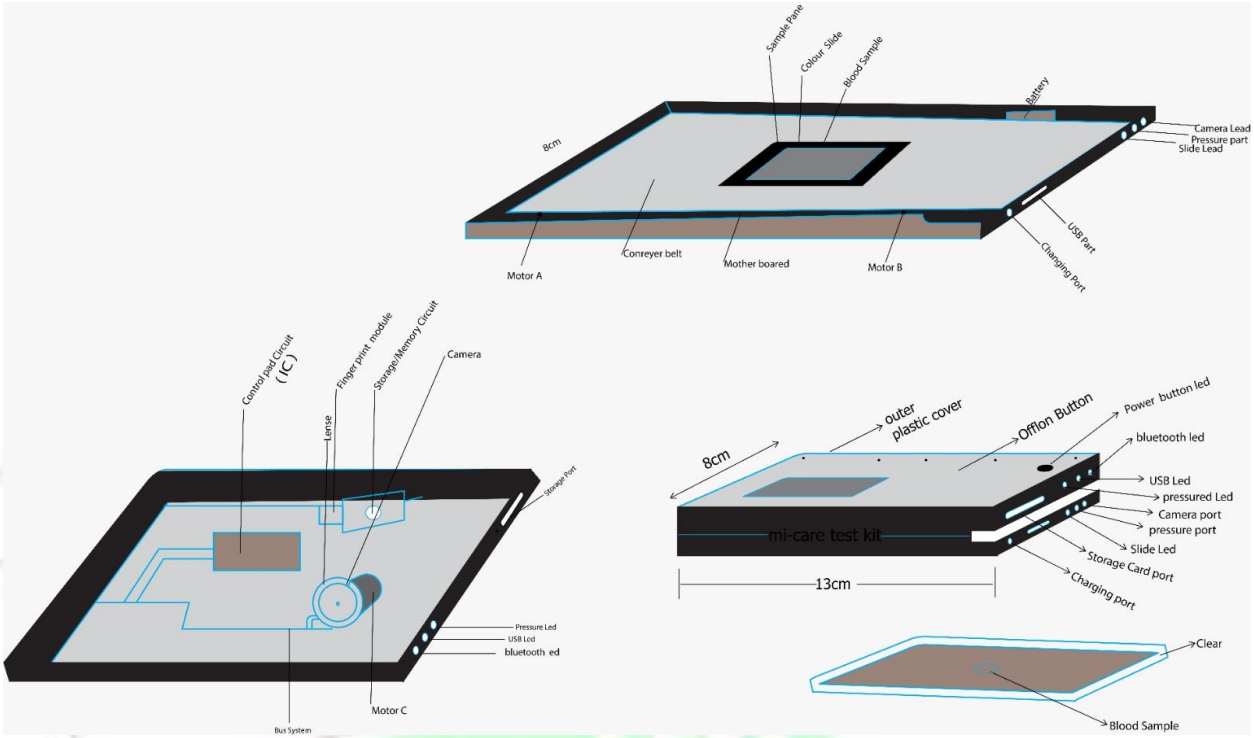
XVI. Subsequent authentications will be as easy as providing your password/PIN and fingerprint after powering on the *Mi_care Test Kit*.



Actual Usage

I. After a successful authentication process, press the slider starter button on the device or on the application. The feeder muzzle should open and present the slider panel upon which you can place your blood sample placed in between two clear slides.
II. The muzzle will never open if the authentication fails.
III. Place your slides with your well prepared samples on the slider panel. Make sure it is centrally positioned on the panel.
IV. Press the slider starter button on the device or the on the application.
V. Adjust the lenses to a clear focal point, adjust the camera to catch finer resolution of the image.
VI. Press validate.
VII. Submit the image for cleaning and preparation for machine Learning Diagnosis.
VIII. Enter the patient's symptoms and Click "*Diagnose*".
IX. Wait for the results, could be a Positive (match) with accuracy levels indicated, Disease name, symptoms and a simple animation of the chemical makeup of the Disease or could be a negative (No match).
X. You may wish to redo the sample preparation or retake image or adjust image size and quality and reenter symptoms then redo the Diagnosis.
XI. Click *Discard and Finish* or *Save and Finish*.

MI_CARE TEST KIT TECHNICAL SPECIFICATIONS.



Block chain Technology and how it will be implemented on Mi_care.

Any patient can be registered or enrolled at any health center of choice. Owing to the heterogeneous and distributed nature of systems as used by various hospitals, the challenge is to collect patient's medical data from every hospital he/she may have registered and still maintain data integrity, consistence and privacy. The Mi_care bridges this by providing a many in one integrated system which can be interfaced by any other hospital system.

The Mi_care e-health management system will be developed on two major emerging technologies; Distributed Databases and Block chain. Distributed systems already exist following the distributed nature of various hospitals. The challenge is maintaining a consisted data record of a single patient at these hospitals.

Block chain is a modern technology which has gained usage ever since the idea was coined in 2009 by *Satoshi Nakamoto*. Block chain can be thought as a digital ledger to store transactional information. The data is stored as signed blocks, which link to each other, creating a chain of immutable interconnected data entries.

To sign a new block, a node needs to find a SHA-256 signature that matches specific criterial. To do so, it will use the nonce field to brute force possible solutions. Any new block needs to be validated with the majority of the validation nodes forming the block chain. Once the block has been validated, it is added to all the nodes of the block chain. If any information in the data inside the block is altered, the signature becomes invalid. To make the block valid again, this signature would need to change. To ensure that the following blocks still work, a new signature would need to be generated for each of them. Even if a node could regenerate those signatures, the changes would need to be accepted by a majority of the nodes hosting the block chain.

For these reasons, block chains are immutable. No information that is included in the data of the blocks can be changed. They are also managed by a set of decentralized nodes, removing the need for a central authority to control all the transactions.

It is for the purposes of the patient's data consistence and integrity across all registered platforms that we will employ block chain technology.
If a patient is admitted at *hospital A* for medication, his/her new medical record has to be reported to any other health center the patient is registered. An agreement has to be reached through a digital vote by the concerned centers before such a record can be added to the block chain. In the case where a digital vote does fail, the record will only be available to the health center the patient has been admitted to. However, the basic information of the patient should be consistent across all the health centers and for this reason any update made on the patient's basic info should only be committed after an agreement has been reached by other hospitals. This may be limited to basic info out of patient's control like updating fingerprints, blood pressure etc.

MI_CARE EHEALTH SERVICE PROTOCOL TECHNICAL SPECIFICATIONS.

DESIGNED BY

Mulongo Duncan | Steven Kamau | Duncan Santiago

MI_CARE EHEALTH SERVICE
www.mi_care.org

Abstract: MI_CARE is a completely peer-to-peer decentralized privacy preserving e-health service and crypto which is built on block-chain technology. Its implementation follows from several cryptographic protocols which are used to achieve anonymity of actors over the block-chain and establishing consensus among participants on the platform while securing the MI_CARE network. Unlike the energy-intensive Proof-of-Work [PoW] protocol as used by Bitcoin cryptocurrency, MI_CARE employs an energy-efficient Proof-of-Stake [PoS] protocol and a lightweight Zero-Knowledge-Proof [ZKP] protocol. The protocols work to secure the MI_CARE network and to preserve the privacy of participants on the block chain. In this way, the MI_CARE is built to serve both as an efficient, accurate and honest e-health service and a cryptocurrency network employing tokenization of health services and using them as a value carrying asset in form of tokens [Hert-Bits]. MI_CARE serves to eliminate the need for health intermediaries by linking clients directly to health services, maintaining medical accuracy through diagnosis and/or prescription validation and allowing participants to transact seamlessly without restriction or borders.

Abbreviations: ZKP, PoS, PoW, PoV, HTB NHS, POPV, PRPV.
Keywords: Agreements, Consensus, Bitcoin, Health service providers, Tokenization.

Introduction: While healthcare stands as a basic need to guarantee human survival, many a time it proofs very difficult for health clients to access quality health services in an efficient manner. And with the surge in the demand of healthcare services in our current world, health service providers have struggled to meet the needs of their clients due to a poorly connected health sector. Many health providers depend on current client's health status and additional fragmented information of the client's medical past in order to provide health services.

In world that is riding on an ever evolving cutting edge technology, there is great need to organize tools that come with modern technology and optimize them for human survival. Rarely have health sectors taken advantage of block chain technology in implementing systems that ensure secure medical data exchange, collective decision among various specialists in the health sector and guarantee pure medical diagnosis, one that is free of human errors and data corruption.
Trust based model as used in current health service arena offers an incomplete implementation of an efficient, affordable, transparent and mostly accurate health service. Although it is very difficult to design a system that can achieve a completely accurate health service, it possible to reduce medical inaccuracies to as low as 5% by use of block chain technology and artificial intelligence.

In existing models, the client must bet on their belief and money by trusting a specific health service provider who may at a times fail to understand the client full and accurate medical past. So far, there exist no solid mechanism that can validate medical diagnosis and/or prescriptions without relying on trusted parties and limited basic principles.

It plausible that there be a way to proof the veracity in the applicability of a given health service which could be some prescription, medication, therapy, rehabilitation etc. Such a service should take into account many factors surrounding the client and not just universal principles of health provision. Cryptographic proofs as applied to block chain could be a game

changer in attaining transparency in providing health services. The need for trust is completely eliminated as trusted parties are dislodged from the model. This new method protects the clients against fraud, misdiagnosis and medical inaccuracies by ensuring that health service decision making process and service dissemination is completely decentralized and distributed among various health specialist and/or health service providers.

Consensus: It refers to a general agreement reached among a group of participants.
In a block chain network, such a process could be formalized by specifying the number of nodes required to agree on the global state of the network for a consensus to be reached.
Consensus Mechanism: It refers to the entire stack of protocols, incentives and ideas that allow a network of nodes to agree on the state of a block chain.

Proof-of-Stake & Proof-of-Veracity: MI_CARE bases its consensus mechanism on Proof-of-Stake protocol whose crypto-economic security involves a set of rewards and penalties applied to capital locked by Stakers. The incentive structure encourages individual Stakers to operate as honest validators, punishes those who don't, and creates an extremely high cost to attack the network.

Proof-of-Stake: Is a way to prove that validators have put something of value into the network that can be destroyed if they act dishonestly. In MI_CARE's Proof-of-Stake, validators explicitly stake capital in form of (HBT) into a smart contract on MI_CARE. The validator is then responsible for checking that new blocks propagated over the network are valid and occasionally creating and propagating new blocks themselves. If they try to defraud the network (for example by proposing multiple blocks when they ought to send one or sending conflicting attestations), some or all of their HBT can be destroyed.

Validators: To participate as a validator, a health service provider must deposit some HBT not less than a predetermined amount into the deposit contract and should be running a software capable of executing a client, consensus and a validator. On depositing their HBT, the health service provider joins an activation queue that limits the rate of new validators joining

the network. Once activated, validators receive new blocks from peers on the MI-CARE network. The transaction delivered in the block are re-executed to check that the proposed changes to MI_CARE's state are valid, and the block signature is checked. The validator then sends a vote (called an attestation) in favor of that block across the network.

Proof-of-Veracity: How can a client verify that a given diagnosis/ or prescription is applicable? Or how can a patient verify and confirm the most efficient medication and/or health routine based on the confirmed diagnosis and/or symptoms?
While Proof-of-Correctness works via loop invariant, which are statements which are true throughout every iteration of the loop, the Proof-of-Veracity achieves the same objective by use of consensus mechanism. Instead of many iteration of loops in a query-response mode, the Protocol sends the query [proposed data block] to as many specialist and/or health service providers as possible.

The veracity of a given block of data is achieved only after a consensus is reached. Unlike the traditional consensus mode which relies on one validator one vote in a discrete sense, Proof-of-Veracity employs a continuous view of attestation from validators. The validator does not limit his validation by issuing a thumbs up or down but goes ahead to explain the reasons behind the vote. In this view, the weight of a given block of data being validated is not only decided by the number of votes issued but also by the similarity in the backing [logical explanations] of such votes. Such attestations are therefore measured in terms of probabilities which indicate how similar a piece of explanations compares to attestations by other validators.

To safeguard the transparency in the attestations, the voting is done in such a way that no validator knows about who the other validators are and what their attestations are. By so doing, the Protocol is fashioned against colluding and cheating thus eliminating any possibility of bias in the attestations.
Proof-of-Veracity is based on the fact that for any diagnosis and/or prescription given by specialists and/or health service providers on the MI_CARE network operating under similar governing principles and/or agreements should replicate same results. Any contradiction is considered as dishonesty. For a specialist and/or

health service provider (A) using mode A(m) with tools A(t) under conditions A(c) should match with the results of specialist (B) using mode B(m) with tool B(t) under conditions B(c).
Hence the equation; *[A, A(m), A(t), A(c) == B, B(m), B(t), B(c)]*.
With arbitrary choice of specialist, Procedures, conditions and tools, the above equation should always hold true for diagnosis and/or prescription to guarantee veracity and validate a block of data on the MI_CARE block chain.
On MI_CARE network, any diagnostic results is treated as a claim until a validation is done by use of the PoV protocol. Hence blocks of data are placed on the unverified queue until they are validated and appended at head of the block chain.

It is possible to make client's symptoms and/or diagnosis public while keeping the client anonymous using the Naked-Hare-Scheme [NHS]. The naked hare decides to cover his face instead of his genitals while his in-laws passes by the river where he is accidently encountered on his bath errands. If he were to cover his genitals and exposed his face, it could be embarrassing for it would be easy for him to be identified and probably shamed, in this mode, the genitals which are considered private are exposed but the identity of the owner is anonymized, the genitals represent client's sensitive medical and/or health records while the face represent the identity of the client. We propose an interactive approach PoV where queries in form of diagnosis and/or prescriptions which form the data block to be validated are gossiped on the network.

In the case of diagnosis and/or prescription via 'Trusted Initiator' and/or "block proposer" who is chosen at random based on the client's preference among validators, we propose a verification step through validators and/or health service providers. The initiator is tasked with creating a new data block which could a prescription and/or a diagnosis. The validator sends out the data block to the block chain where the smart contract assigns it to authenticated validators who may be chosen at random from those who qualify to perform validation. Different data blocks will need different validators chosen based on the type of health services they can offer.
There could be a witness in form of an AI which could be implemented in the smart contract. The witness acts through a smart contract and serves as a moderator which compares

attestations from validators and assign them variability ratios. The more close the response are, the stronger the consensus. The POV protocol is designed to test for consensus while the Witness serves to confirm such tests of consensus. An AI system may have access to a particular client's medical data and other records of interest. Such an access is authorized and secured through Public-Private key encryption scheme such that no one, even the validators can snoop on the data being authorized to an AI. The AI preserves a copy of the entire block chain state just like other nodes does. In the case where the client proposes and/or initiates a new block of data which could be symptoms which seeks diagnosis and prescriptions, insurance requests, payments etc. Such a request is gossiped directly on the network without the need of a random initiator. By saying so, we imply that a client as so long he/she is certified on the network can work as a block proposer. He or She may not necessarily be a validator in the literal sense but there exist contribution from his node in such a way that he can authorize a witness, the AI, to access his health records to be used while validating data blocks.

Several steps could be involved during the POV test and consensus confirmation.

1. A comparison among attestations of validators is performed and variability ratios and/or probabilities [S_0/S_x] calculated based on how attestations vary from each other. This is done by use of an AI [Witness] implemented in a smart contract based on given agreements and conditions. This will be discussed further.
2. A comparison of each validator's attestation is compared to the ledgers' verified attestations. [A_s/A_l].
3. The Average attestations' value by validators is compared to the ledgers' verified attestations. [A_v/A_l].
4. Each validators' attestations is compared to the average of validator's committee attestations. [A_s/A_v].
5. Accuracy levels are calculated from the above factors by a witness that leverages and balances facts on the ledger, client's records and validators' attestations.
6. Block validation completion. The proposed block may be validated and appended to the block chain head. It fails to be validated, an entirely new block could be generated during

the validation process basing its argument from the client's requests. This is done by coupling many factors surrounding validators and the client's health record and agreements by the smart contract. Once this is completed, the recommendation is send to the client to pursue further actions which could be treatment.

Validators on the block chain are completely anonymous and their attestations are kept private to avoid collusion and eliminate bias. The smart contract in tasked with performing the above steps.

Zero-Knowledge-Proof: A zero knowledge proof is a way of proving the validity of a statement without revealing the statement itself. The "prover" is the party trying to prove a claim, while the "verifier" is responsible for validating the claim.

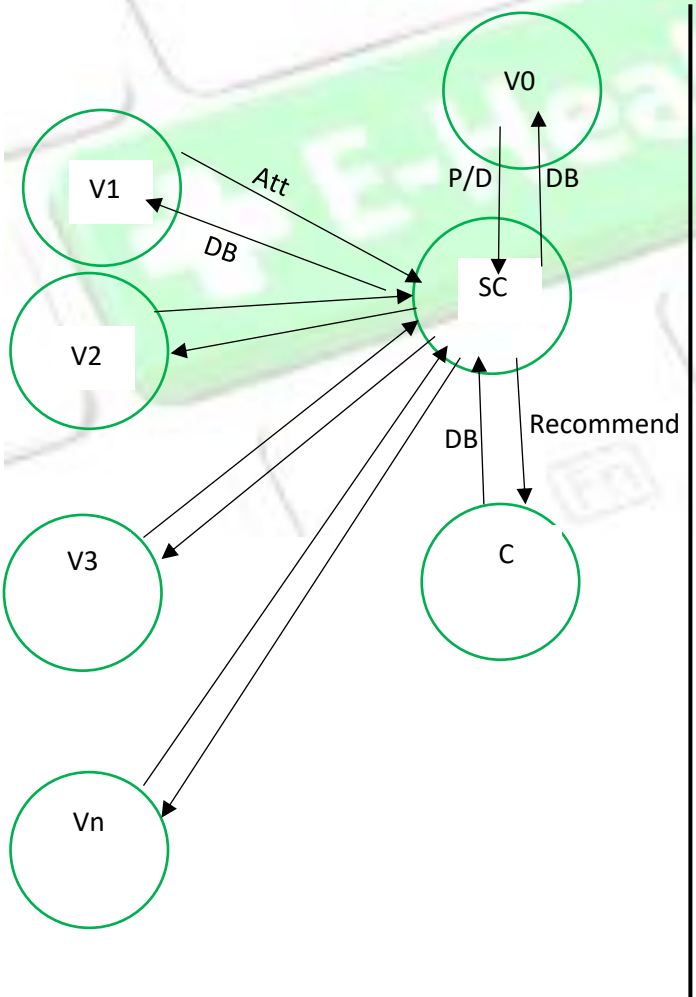
In the case where some validators may want verification on how the validation has been performed and how the consensus have been reached, the AI will employ the ZKP to convince the validators that a particular verified block is valid by confirming queries given via an interactive approach where the veracity of a given data block is given without revealing the client's personal data or other validators attestations. Equipped with data on the Block Chain and client's health records, the AI can determine the probability of a person having disease X. This has to be verified and voted for. As much as an AI can serve as a witness comparing the attestations and establishing consensus among validators, it can also serve as a block proposer based on the symptoms given and other diagnostics.

Contradictory attestations attracts penalties while consistence attestations leads to a reward in form of HBTs. This among others will be discussed later while detailing Tokenization protocol as it will be used on MI_CARE network.

Efficiency: We propose a delayed validation approach where results of a given diagnosis and/or prescription are not guaranteed to be instant. A track record of both validated and yet-to-be-validated data blocks is kept. By default, gossiped data blocks will be added to the yet-to-be-validated block and after they are verified, they are appended to the block chain head.

In the case of an emergency, one could be attended to and prescribed in whichever mode by an authenticated validator but the diagnosis and/or prescription given may be gossiped on the MI_CARE network as a new data block which requires validation, this is

Post Prescription Validation, [POPV]. This seeks to reinstate a particular prescription or falsify it.
In the case of a risky medication, the most efficient prescription should be determined before such a medication is authorized, this is Pre-prescription Validation, [PRPV]. This seeks to confirm the safety of a given prescription and applicability of a given prescription.
Tokenization: Based on Proof-of-Stake and the PoV, we propose a tokenization protocol in which service providers and clients can transact by use of a token, HBT, (a value holding asset native to MI_CARE). Such a token could be mostly stable due to unchanging nature of health services. Before any client, health service provider is certified to become a block proposer and/or validator on the MI-CARE network, he/she must deposit a predetermined amount in HBTs on MI_CARE. Any successful validation of a data block is rewarded in form of HBTs and any dishonesty acts are penalized by destroying some or even all staked HBTs of a given validator and/or block proposer.



KEY

The relationship between validator V0, Smart contract SC and Client C is in this case indirect data block proposition.

Eliminating V0, becomes a direct data block proposition which involves the client and the block chain without going through a Trusted Initiator, V0.

V: Validator

SC: Smart contract with AI

C: Client

Att: Attestation (Vote), DB: Data block,

Note: How payment is done has not be shown on the diagram.

We have only considered a single service involving a client and a medical service provider. This can be generalized to cover other services like gym, insurance, nutrition Etc.

This doc may contain unintentional errors

Abstract: Medical care has become one of the most indispensable parts of human lives, leading to a dramatic increase in medical big data. To streamline the diagnosis and treatment process, healthcare professionals are now adopting Internet of Things (IoT)-based wearable technology. Recent years have witnessed billions of sensors, devices, and vehicles being connected through the Internet. One such technology—remote patient monitoring—is common nowadays for the treatment and care of patients. However, these technologies also pose grave privacy risks and security concerns about the data transfer and the logging of data transactions. These security and privacy problems of medical data could result from a delay in treatment progress, even endangering the patient's life. We propose the use of a blockchain to provide secure management and analysis of healthcare big data. However, blockchains are computationally expensive, demand high bandwidth and extra computational power, and are therefore not completely suitable for most resource-constrained IoT devices meant for smart cities. In this work, we try to resolve the above-mentioned issues of using blockchain with IoT devices. We propose a novel framework of modified blockchain models suitable for IoT devices that rely on their distributed nature and other additional privacy and security properties of the network. These additional privacy and security properties in our model are based on advanced cryptographic primitives. The solutions given here make IoT application data and transactions more secure and anonymous over a blockchain-based network.

Keywords: blockchain; medical big data; Internet of Things; smart contract; Ethereum; data preservation; key management; authentication; ring signature; smart cities

health data to smart contracts, a smartphone with internet connectivity and an RPM application

(Figure 1). Wearable devices and IoT play an important role in RPM and in the current push to develop Smart Cities. Wearable devices collect patient health data and transfer it to hospitals or medical institutions to facilitate health monitoring, disease diagnosis, and treatment. In doing so, we see a Big Data situation develop through all the patient data being analyzed and transferred.

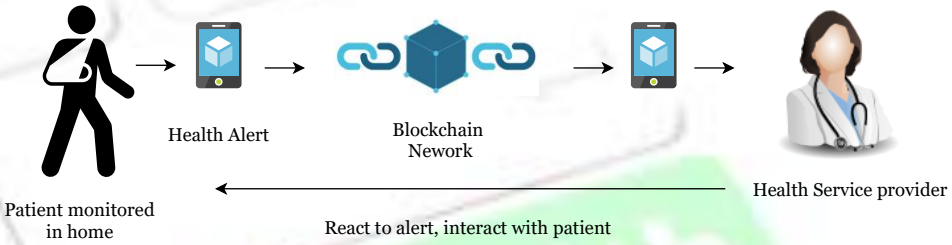


Figure 1. Remote Patient Monitoring.

Wearable devices in healthcare are smart electronic devices with micro-controllers that can be embedded into clothing or worn on the body as accessories. They are unobtrusive, user-friendly, and connected with advanced features such as wireless data transmission, real-time feedback and alerting mechanisms built into the device. These devices can provide important information to healthcare providers such as blood pressure, blood glucose levels and breathing patterns just to name a few. Healthcare devices can be categorized into four types (Figure 2):

- Stationary Medical Devices–devices can be used on a specific physical location (e.g., chemotherapy dispensing stations for home-based healthcare)
- Medical Embedded Devices–devices which can be implanted inside the body (e.g., pacemakers)

- Medical Wearable Devices–prescribed devices by doctors (e.g., insulin pump) • Wearable Health Monitoring Devices–consumer products (e.g., Fitbit, Fuelband, etc.)

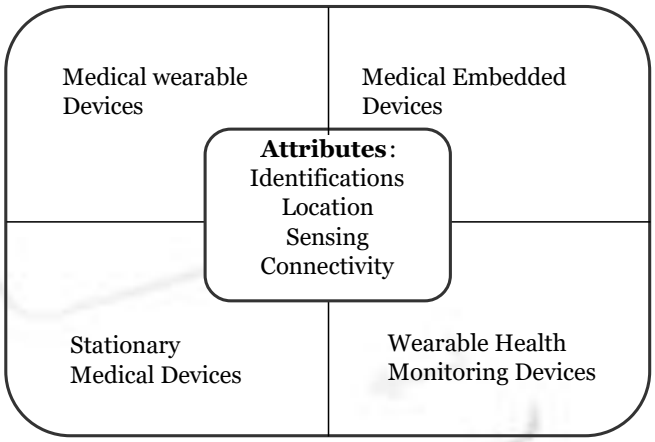


Figure 2. Healthcare IOT Topology.

On 13 November 2017, the Food and Drug Administration (FDA) approved the first pill with a sensor inside of it (aripiprazole tablets with sensor) that can track if a patient has swallowed it. This pill's sensor sends messages to a wearable patch, and the patch itself transmits the message to a mobile application on the smartphone. This technology could be a game changer for chronic disease and mental health disorders.

One of the facets of the Internet of Things (IoT) is the network of wearable devices, embedded with software, electronics, sensors, actuators, and connectivity which enables the wearable device to connect and exchange data (Figure 3). In a futuristic smart city, we will not only see these wearable devices transmitting healthcare data, but it is reasonable to assume that wearable devices can share a myriad of data as we interconnect these devices. Therefore, the reach of the ideas presented here regarding wearable healthcare devices and using blockchain technology are further reaching than we show or can imagine here.

To handle such patient data with other institutions, such infrastructure demands secure data sharing. Health data is highly private and sharing of data may raise the risk of exposure. Furthermore, the current system of data sharing uses a centralized architecture which requires centralized trust.

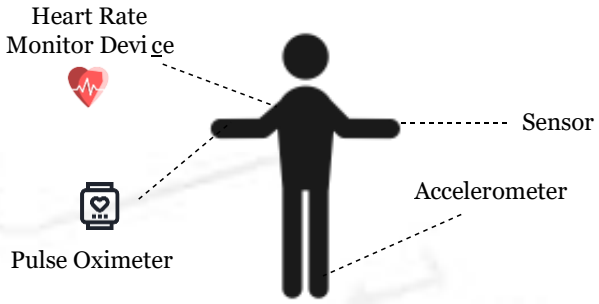


Figure 3. Wearable devices for the patient.

The solution for data privacy and security could and should very well be blockchain technology. Initially proposed by Satoshi Nakamoto in Ref. [1], blockchain technology provides the robustness against failure and data exposure. The blockchain is a shared data structure responsible for storing all transactional history. The blocks relate to each other in the form of a chain. The first block of the chain is known as Genesis. Each block consists of a Block Header, Transaction Counter and Transaction. It acts as a decentralized architecture to record the data. The structure of blockchain is summarized in Table 1.

Table 1. Structure of the Blockchain.

Field	Size
Magic Number	4 bytes
Block Size	4 bytes
Header: Next 80 bytes	
Version	4 bytes

Previous Block Hash	32 bytes
Merckle Root	32 bytes
Timestamp	4 bytes
Difficulty Target	4 bytes
Nonce	4 bytes
Rest of Blockchain	
Transaction Counter	Variable: 1 to 9 bytes
Transaction List	Depends on the transaction size: Upto 1 MB

Blockchain security is based on a *proof of work* concept, and a transaction is only considered valid once the system obtains proof that a enough computational work has been exerted by authorizing nodes. The miners (responsible for creating blocks) constantly try to solve cryptographic puzzles (named Proof of Work (PoW)) in the form of a hash computation. The process of adding a new block to the blockchain is called *mining*. Each block in the chain is identified by a hash in the header. The hash is unique and generated by the Secure Hash Algorithm (SHA-256). SHA takes any size plaintext and calculates a fixed size 256-bit cryptographic hash. Each header contains the address of the previous block in the chain. The inability to delete or change information from blocks makes the blockchain the best appropriate technology for the healthcare system. However, adopting blockchain in the context of IoT is not straightforward and entails several problems such as demands of high computational power to solve PoW, low scalability and long latency for transaction confirmation over the network.

We propose a novel model of blockchain and eliminate the concept of PoW to make it suitable for IoT devices. Our model relies on the distributed nature and other additional security properties to the network. Transactions in blockchain are broadcasted publicly in the blockchain network and contain additional information about both the sender and recipient. If we talk about the blockchain that underpins bitcoin, everyone has a public address, and anyone can see what funds they already have stored at that address. This way, users cannot be anonymous in the network. For anonymity and the authenticity of the user, we present a lightweight privacy-preserving ring signature scheme that is suitable for anonymous transactions by authentic users.

On the one hand, a lightweight digital signature guarantees that information has not been modified, as if it were protected by a tamper-proof seal that is broken if the contents were altered. On the other hand, a ring signature [2] allows a signer to sign a message anonymously. The signature is mixed with other groups (named ring) and no one (except actual signer) knows which member signed the message. We are not using heavy operations such as pairing and exponentiation in ring signature to make it more suitable for blockchain and IoT.

We also use double encryption of data using lightweight encryption algorithms (ARX ciphers) and public encryption schemes. Here double encryption means, we firstly encrypt the data using symmetric key encryption and then we encrypt the symmetric key itself using a public key. Please note that, here we are not encrypting the same data twice with different keys. ARX is a class of cryptographic algorithms which uses three simple arithmetic operations: namely modular addition, bitwise rotation, and exclusive-OR. In both industry and

academia, ARX cipher has gained a lot of interest and attention in the last few years. For securely exchanging cryptographic keys over a public channel, we use the Diffie–Hellman key exchange technique. Using both these techniques together will guarantee the security, privacy and anonymity of user's data using lightweight techniques suitable for small IoT devices.

Related Work

In this paper, we pull some of our main motivation to explore blockchain in healthcare from Refs. [3–8], where the respective authors systematically mentioned some of the latest trends in blockchain research. Since the introduction of Bitcoin in Ref. [1], the possibilities are endless of how the underlying technology can be used in other ways outside of the financial realm. There have been numerous attempts at applying blockchain technology outside of the financial realm [9–14]. It is easy to imagine the far-reaching applicability of this technology specifically in healthcare, smart cities and IoT.

In Ref. [15], there was an introduction to methods for using blockchain to provide proof of pre-specified endpoints in clinical trials. Irving and Holden empirically tested such an approach using a clinical trial protocol where outcome switching had previously been reported. They confirmed the use of blockchain as a low cost, independently verifiable method to audit and confirmed the reliability of scientific studies. We use lightweight digital signature schemes in our model inspired by Ref. [2].

We have to date already heard of many data breaches or data losses with regards to medical data [16,17]. Health information is something hackers will seek out as it may contain pertinent information for identity theft. Medical record ownership is another key point when

discussing health information. The records themselves come in many forms, reports, images, videos, and raw data. They could potentially also come in many different formats depending on the systems in use by the given provider. The integrity of these records then becomes paramount. There needs to be contingencies in place to ensure the integrity of the data is maintained to ensure the data has not been changed, destroyed, or removed. The access to the data should be controlled by the patients; however, they themselves should not be able to alter it either. The patient records should be consistent and available across institutional boundaries [18,19].

In the context of Smart cities and Big Data, we have seen some strong work recently by Wu and Ota in Refs. [20–23]. They have really been able to focus on how IoT, Smart Cities, and the resulting Big Data are all important factors going forward with Smart City design and implementation. We have contributed some work already in cryptanalysis of ARX ciphers and other security algorithms [24–30].

Drawbacks and Security Issues

The main concern in RPM systems is the secure and efficient transmission of the medical data. Healthcare data is a lucrative target for hackers and therefore securing protected health information

(PHI) is the primary motivation of healthcare providers. Healthcare has become the primary target for cybercriminals. For example, cyber-attacks on medical devices or health data have become more common in the last decade. However, the inability to delete or change information from blocks makes blockchain technology the best technology for the healthcare system and could prevent these issues. However, blockchain technology in

its original form is not enough of a solution. In this section, we discuss the challenges for applying blockchain to the IoT and explain how to solve these problems

in our model.

System Requirements and Our Solutions

1. Decentralization: To ensure robustness and scalability and to eliminate many-to-one traffic flows we need a decentralized system. Using such decentralized systems, we can also eliminate the single point of failure or information delay problems. In our model, we are using an *overlay* decentralized network.
2. Authentication of data: User's computer or cloud services store unpreserved data that needs to be transferred to blockchain networks. During transmission, the data could be modified or lost. The preservation of such incorrect tampered data increases the burden to the system and can cause the loss of the patient (death). Therefore, to ensure that data is not modified, we use a *lightweight digital signature* [2] scheme. On the receiver side, data is verified with the user's digital signature, and if received correctly, it sends a receipt of data to the patient.
3. Scalability: Solving PoW is computationally intensive; however, IoT devices are resource restricted. Also, the IoT network contains many nodes and blockchain scales poorly as the number of nodes in the network increases. We eliminate the concept of PoW in our overlay network and divide our overlay network into several clusters instead of a single chain of blocks, and therefore a single blockchain is not responsible for all nodes. Instead we spread the nodes over several clusters. Our model relies on the distributed nature and other additional security properties to the network.

4. Data Storage: Storing IoT big data over blockchain is not practical and therefore we use cloud servers to store encrypted data blocks. The data is safe over the cloud due to additional cryptographic security like the digital signature and high standard encryptions which will be discussed later. However, it may cause a problem about trusted third parties. For this purpose, we store all transactions in different blocks and create a combined hash of each block using Merkle Tree and transfer it to the distributed network. This way, any changes in cloud data can be easily detectable. Doing the storage in this manner also preserves the decentralization over some extents.
5. Anonymity of users: Medical data of a patient may contain sensitive information, and therefore data must be anonymized over the network. For anonymity, we are using lightweight Ring structure [2] along with digital signatures. *Ring signature* allow a signer to sign data anonymously, that is the signature is mixed with other groups (named ring), and no one (except actual signer) knows which member signed the message.
6. Security of data: Medical devices or health data must be accurate and cannot be changed by hackers. To save the data from hackers, we are using a double encryption scheme. Here double encryption does not refer to encrypting the same data using two keys but instead encryption of the data and again encryption of key which was used to encrypt data. We encrypt the data using lightweight *ARX algorithms* and then encrypt the key using the public key of the receiver. Also, we are using the *Diffie–Hellman key exchange* technique to transfer the public keys and therefore getting the keys is almost impossible for an attacker.

Our System

Our system consists of five parts: Overlay network, Cloud storage, Healthcare providers, Smart contracts and Patient equipped with healthcare wearable IoT devices.

1. Cloud Storage: Instead of saving the IoT healthcare data over blockchain, we use cloud storage servers to save the patient data. The cloud storage groups user's data in identical blocks associated with a unique block number. These clouds are connected to overlay networks, once the data stored in a block, the cloud server sends the hash of the data blocks to the overlay network. The hash of the data in the single block is calculated using Merkle Tree (Figure 4). If the overlay network accepts the root hash of the new block, it adds the new hash with the previous hash value and generates the new hash of the chain. In such cases, we do not need any third-party trust, because any changes in data could be easily traceable.
2. Overlay network: An overlay is a peer-to-peer network that is based on distributed architecture. The nodes connected to the network could be a computer, smartphone, tablet or any other IoT device as well (Figure 5). (Please note that, in the description of overlay networks we assume that readers have sufficient knowledge of standard cryptographic protocols and use of the hash function in bitcoin mining.) In our model, a network consists of specific nodes and they need to prove that they are certified with a valid certificate. Such a certificate can be uploaded or verified before making an account on the network. Once authorized, he/she will be able to sign data/transaction over the network digitally. To increase network scalability and avoid network delay, we group the nodes in the form of many clusters. Each cluster has one Cluster Head that

takes care of public keys of the nodes. Any node attached with any cluster can change the cluster at any time in case of delay. Also, the nodes attached to a cluster can change the cluster head. Cluster head maintains the public keys of requesters (healthcare providers), who can access the data of a particular patient, and the public key of requestees (patient) that are allowed to be accessed.

Consider the case where a patient wants to share his/her data with a particular doctor, then the node digitally signs the transaction and sends it to the network with a public address of the doctors' node. The cluster head verifies the patient digital signature and patient public key, and if it is verified correctly, cluster head searches the public key of the doctor node in his own cluster. If the public key is available, then it will broadcast the transaction to its own cluster, and if doctor nodes public key is not available then cluster head will broadcast the transaction to other clusters. In the case where the digital signature or public key of any node is not verified then the cluster will not broadcast data in its cluster but transfer transaction to other cluster heads. Cluster heads are also responsible for storing the hash of the data block stored in the cloud. Each new block in the cluster contains the hash of the previous block also (Please note that each hash block says $hash_n$ is combined hash of all previous hashes such as $hash_1$, $hash_2$..$hash_{(n-1)}$). A cluster head can independently decide whether to keep hash of new data block or not. Once a cluster head adds new hash it will broadcast this to all clusters. Other clusters also verify the new block using the hash value of the previous chain. To follow the distributed trust in the network, each cluster head maintains a trust rating for other cluster head

based on Beta Reputation System [31]. For more details of overlay networks we suggest readers to reference the following papers [4,32].

3. Healthcare providers: Healthcare providers are appointed by insurance companies or by patients to perform medical tests or to provide medical treatments. Healthcare service providers deal with treatment of patients once they receive an alert from the network. They are also treated as a node in the network and authorized to receive particular patient data from the cloud.
4. Smart contracts: Smart contracts allow the creation of agreements in any IoT devices which is executed when given conditions are met. Consider we set the condition for the highest and lowest level of patient blood pressure. Once readings are received from the wearable device that do not follow the indicated range, the smart contract will send an alert message to the authorized person or healthcare provider and also store the abnormal data into the cloud so that healthcare providers can receive the patient blood pressure readings as well later on if needed.
5. Patient with wearable IoT devices: The IoT device will collect all health data from the patient. Such data could be heartbeats, sleeping conditions, or walking distance to name a few. Patients themselves are the owners of their personal data and responsible for granting, denying or revoking data access from any other parties, such as insurance companies or healthcare providers. If the patient needs medical treatment, he/she will share personal health data with the desired doctor. Once the treatment is finished the patient can deny further access to the doctor, healthcare provider or health insurance company.

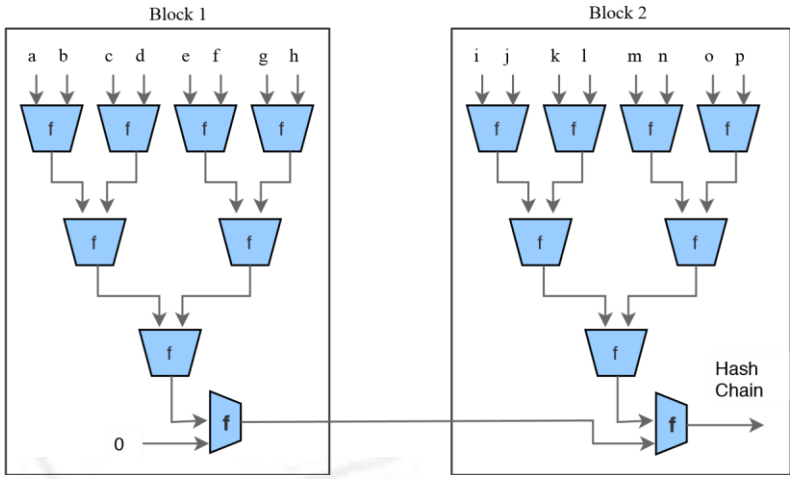


Figure 4. Merkle Tree.

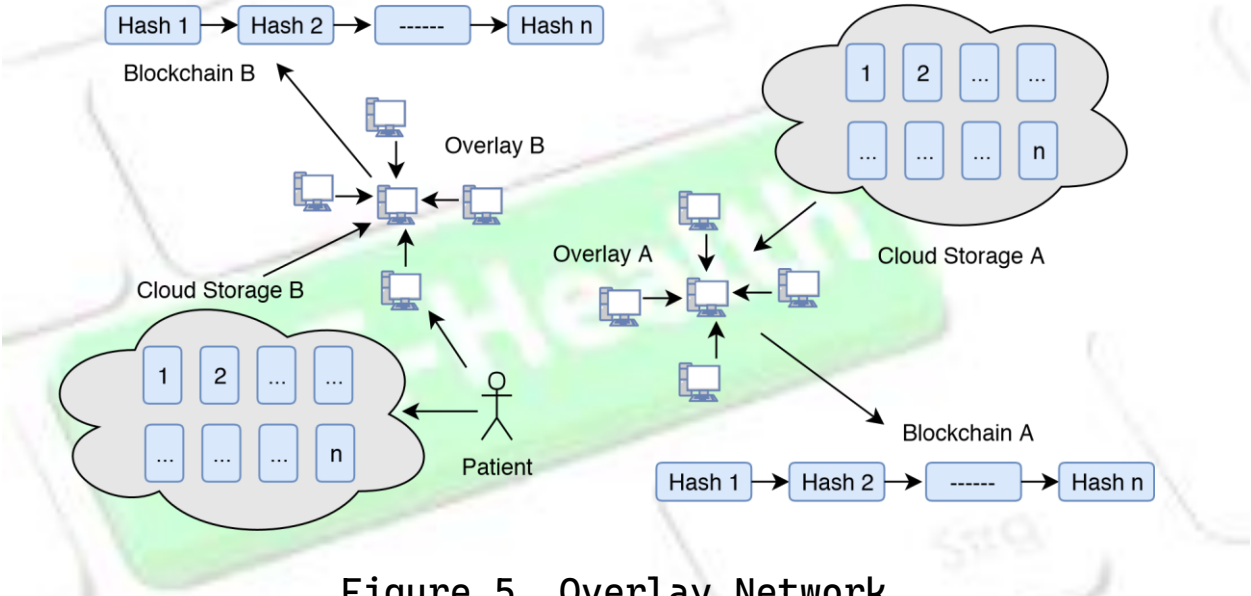


Figure 5. Overlay Network.

Cryptographic Techniques Used in the Model

Instead of only one type of encryption technique, we use both encryption schemes, namely Symmetric and Asymmetric for different purposes. A symmetric algorithm (Private key encryption), as shown in Figure 6, uses the same key for both encryption of plaintext and decryption of ciphertext,

whereas asymmetric algorithms (Public key encryption) use different keys for encryption of plaintext and decryption of ciphertext. We use the variable name k_{sym} for the private key or symmetric key in our algorithms,

and the same key will be used for encryption and decryption on both side of the transmission.

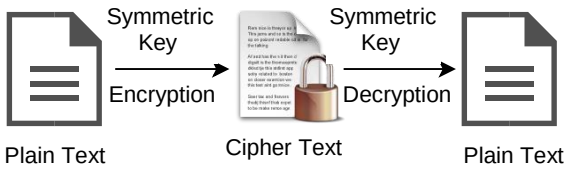


Figure 6. Symmetric Key Encryption.

In the case of asymmetric encryption, sender will have one key pair sk_{priv}, sk_{pub}, and receiver will have another key pair rk_{priv}, rk_{pub}, shown in Figure 7. Data can be encrypted using receiver's public key rk_{pub} and can be decrypted using their private key rk_{priv}. Generally, we use abbreviations plaintext (P) for the unencrypted data and ciphertext (C) for the encrypted data.

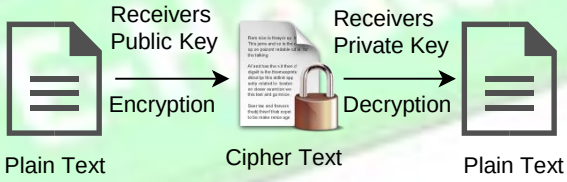


Figure 7. Asymmetric Key Encryption.

5.1. ARX Encryption Algorithm

In our model, we are using a particular branch of Symmetric key encryption, called ARX algorithms to encrypt the data for blockchain. These algorithms are made of simple operations Addition, Rotation and XOR and support a lightweight encryption for small devices. Among a few well known examples, the one example of latest usage of ARX cipher is: SPECK [33], designed by the National Security Agency (NSA), of the United States of America (USA) in June 2013. However, SPECK itself has been severely criticized prior to ISO standardization rejection due to the possibility of the well known

cipher backdoor issue, but still, we use it here because it is safe against key recovery attacks. Our model is specifically dedicated to securing the network against various attacks rather than to secure individual nodes. In the case where within the network a defaulter node is found, we can automatically block it. SPECK is a family of lightweight block ciphers with the Feistel-like structure in which each block is divided into two branches, and both branches are modified at every round. We show the round function of SPECK in Figure 8. Each block size is divided into two parts, the left half and right half.

SPECK Round Function

SPECK uses 3 basic operations on n-bit word for each round:

- bitwise XOR, $\oplus$,
- addition modulo 2^n,
- left and right circular shifts by r_2 and r_1 bits, respectively.

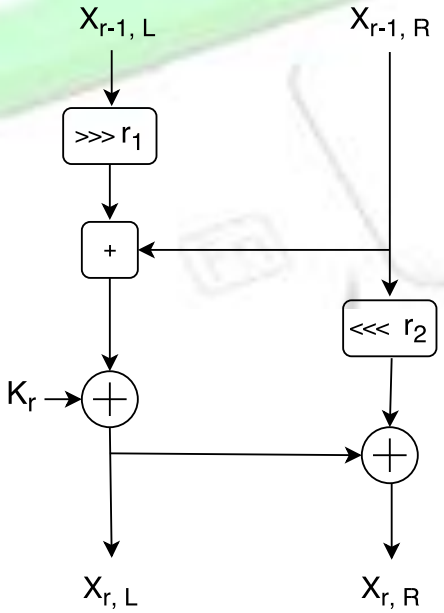


Figure 8. The round function of SPECK.

The left half n-bit word is denoted by $\chi_{r-1,L}$ and the right half n-bit word is denoted by $\chi_{r-1,R}$ to the r-th round and n-bit round key applied in the r-th round is denoted by k_r. $\chi_{r,L}$ and $\chi_{r,R}$ denotes output words from round r which are computed as follows:

$$\chi r,L = ((\chi r-1,L \gg r1) \quad (1)$$
$$\chi r-1,R) \oplus kr$$

$$\chi r,R = ((\chi r-1,R \ll r2) \oplus$$
$$\chi r,L) \quad (2)$$

Different key sizes have been used by several instances of the SPECK family and the total number of rounds depends on the key size. The value of rotation constant r_1 and r_2 are specified as: $r_1 = 7$, $r_2 = 2$ or $r_1 = 8$, $r_2 = 3$ for various variants of SPECK.

5.2. *Digital Signature*

We add a digital signature to the data for authentication purposes. However, applying normal digital signatures is not suitable due to resources limit in IoT devices. Therefore, we suggest using lightweight digital signatures suitable for small devices as given in Ref. [2]. Digital signatures are the public key primitives of message authentication. Each user has a public-private key pair. Generally, the key pairs used for signing/verifying and the key pairs used for encryption/decryption are different. In our case here, sender will have one key pair sk_{spriv}, sk_{spub}, and receiver will have another key pair $rkspriv$, $rkspub$.

The senders private key sk_{spriv} is used to sign the data, and the key is referred to the signature key while senders public key sk_{spub} is used for verification on the receiver side of the transmission. Signer feeds the data or plaintext into the *Hash Function* and generates the hash value $hash_p$. Hash value $hash_p$ of plaintext and signature key sk_{spriv} are then fed to the signature algorithm and sent along with the encrypted data (Figure

9). During the verification process, the verifier generates the hash value $hash_r$ of the received data from the same hash function. Using the Verification algorithm and signers public key, he/she also extracts the original hash value $hash_p$ of plaintext and if the value of $hash_p$ and $hash_r$ are the same then the data is verified and not changed during the transmission process.

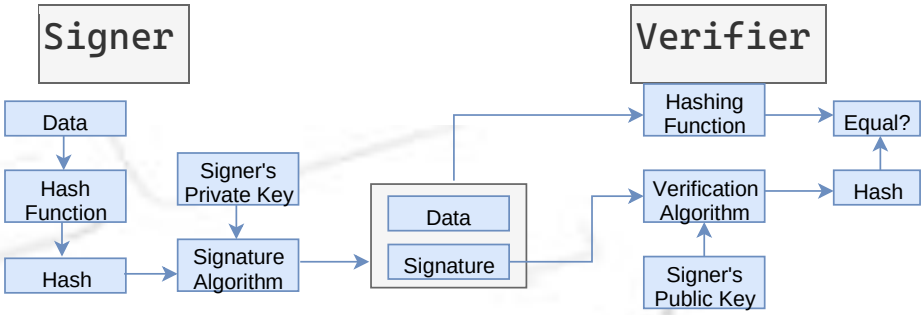


Figure 9. Digital Signature.

5.3. *Digital Ring Signature*

We use lightweight *Ring signature* technology [2] which allows a signer to sign data in an anonymous way (Figure 10). That is the signature is mixed with other groups (named ring) and no one (except the actual signer) knows which member signed the message. Ring Signature was originally proposed by Rivest in 2001 [34]. A user desiring to mix his transaction sends a request to the blockchain network. The request comprises the public key sk_{spub}. After receiving the request the network sends back a certain amount of public keys $sk1_{spub}$, $sk2_{spub}$, $sk3_{spub}$, $sk4_{spub}$ which are collected from other users ($u1$, $u2$, ..., uN) who also applied for mixing service, including sk_{spub}. Using ring signature in our model we can get two important security properties. We achieve both *Signers Anonymity* and *Signature Correctness*.

1. Signature Correctness: A valid signature is always accepted, and an invalid signature is always rejected.
2. Signers Anonymity: A signature is produced by one member from the set of public key holders. Therefore,

the identity of the signer is always hidden in the network, and no one can find out who is the real signer from the signature.

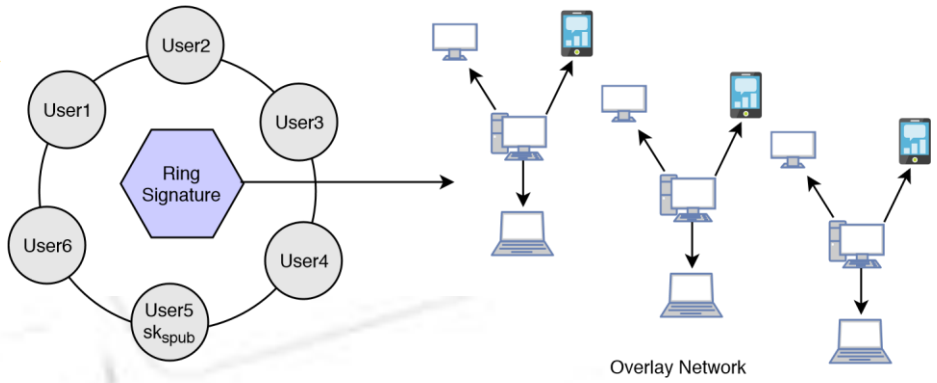


Figure 10. Ring Signature.

5.4. Diffie–Hellman Key Exchange

In all our previous techniques proposed, we need to transfer the public key through the network. To make data more secure, we also share the public key secretly. To share the public key sk_{spub} safely along the network we are using the Diffie–Hellman key exchange technique. A basic Diffie–Hellman exchange of a shared secret between Alice and Bob could take place in the following manner:

1. Alice and Bob generate their own private/public keys (k_A; K_A) and (k_B; K_B). Both publish or exchange their public keys and keep the private keys for themselves.
2. Clearly, it holds that

$$S = k_A \cdot K_B = k_A \cdot k_B \cdot G = k_B \cdot k_A \cdot G = k_B \cdot K_A \quad (3)$$

Alice could privately calculate $S = k_A \cdot K_B$, and Bob $S = k_B \cdot K_A$, allowing them to use this single value as a shared secret. For example, if Alice has a message m to send Bob, she could hash the shared secret $h = H(S)$, compute $x = m + h$, and send x to Bob. Bob computes $h' = \mathcal{H}(S)$, calculates $m = x - h'$, and learns m.

An external observer would not be able to easily calculate the shared secret due to the DLP

(discrete logarithm problem), which prevents them from ending k_A or k_B. Since the output of hash functions is 'random', the message m is information-theoretic secure from adversaries who know x, K_A, and K_B.

Algorithms to Implement Cryptographic Techniques between Sender and Receiver in Our Model

In our encryption Algorithm 1, we encrypt the *data_file* by using the symmetric key k_{sym} and produce a ciphertext file C. After encryption, we use double encryption technique and encrypt the key k_{sym} by using public key cryptography. We use the receivers public key rk_{pub} to encrypt the symmetric key k_{sym} and send the encrypted key along with the ciphertext C. We denote the encrypted symmetric key with C_k.

Algorithm 1 Data Encryption.

1: function ENCRYPTION (*data_file*)
2:if user confirm data preservation over blockchain then
3:Generate a symmetric key k_{sym}
4:$C \leftarrow Encrypt_{sym}$ (*data_file*, k_{sym})
5:$C_k \leftarrow$ *Encryptasym* (*ksym*, rk_{pub})
6: else
7: Do nothing
8: end if
9: end function

For the digital signature senders can use two keys sk_{spub}, sk_{spriv} that is different from the encryption/decryption keys. To add the digital signature, the sender first passes the data file to the

Hash Function and creates the hash value $hash_p$ of the data. Then he/she signs the data using his/her private key sk_{spriv} by passing the value of the private key and hash value $hash_p$ to the Signature Algorithm. The signers public key sk_{spub} can be used to verify data on the receiver's side. To apply the Anonymity of the patient or user, we add ring signature in our Algorithm 2. The user will ask the network for other accounts who also want to add ring signature to their transactions. The network will provide him/her a set of users who also wish to apply ring signature. The sender's transaction is then mixed with other users' transactions and then send it over the network. No one will be able to identify the original signer of the ring group. The process is described in the block diagram (see Figure 11) of our model.

Algorithm 2 Ring Signature and Public Key Sharing.

1: function SIGNATURE (*data_file*)
2:if user chose anonymity over blockchain then
3:Generate an asymmetric public-private key pair sk_{spub}, sk_{spriv}
4:$hash_p$ ← calculate hash of the *data_file*
5:Create the Digital Signature using $hash_p$ and signers private key sk_{spriv}
6:Share the public key sk_{spub} to the receiver using Diffie-Hellman key exchange
7:Mix the signature with another network group to form a ring
8: end if 9: end function

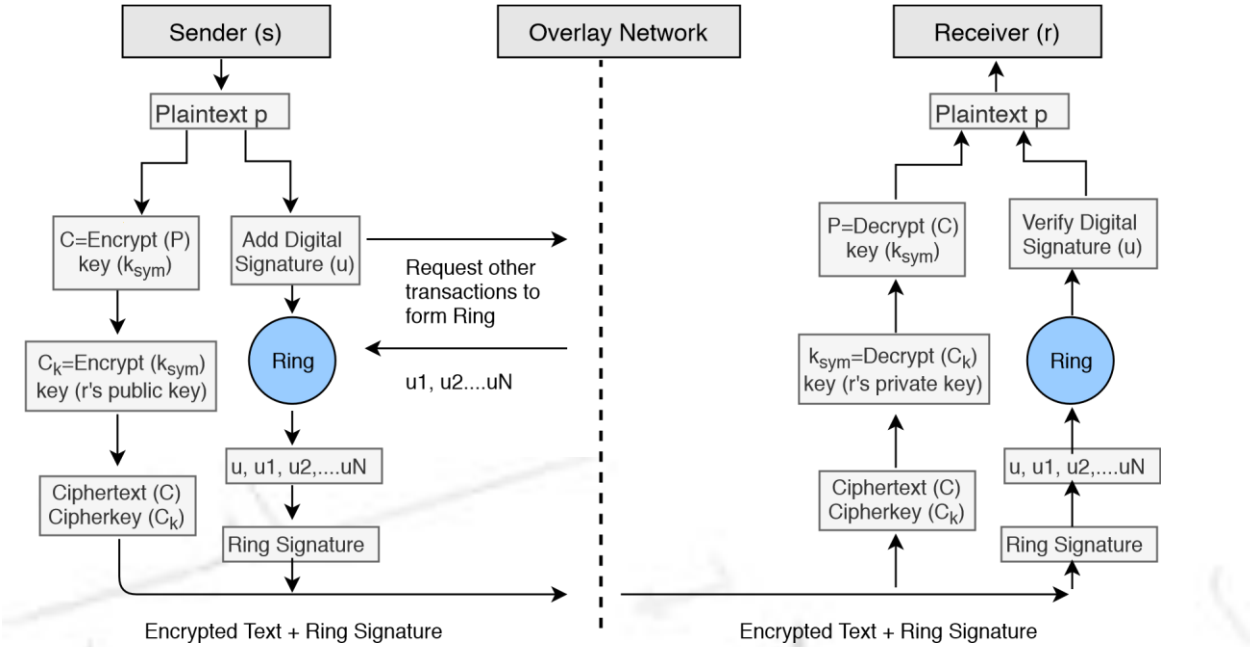


Figure 11. Block Diagram of Model.

To decrypt the ciphertext data C (Algorithm 3), we need the symmetric key k_{sym}. The symmetric key was encrypted using the public key rk_{pub} of the receiver, and therefore receivers private key rk_{priv} can only decrypt the symmetric key. We firstly decrypt the C_k using the private key rk_{priv} of the receiver and get the original symmetric key k_{sym}. We apply the key to the ciphertext C and get the original plaintext or data file.

Algorithm 3 Data Decryption.

1: Input: Encrypted file C, Encrypted symmetric key (C_k)
2: Output: Decrypted $data\_file$
3: function DECRYPTION (C, C_k, rk_{priv}, k_{sym})
4: $ksym \leftarrow Decryptasym\ (C_k, rk_{priv})$
5: $data\_file \leftarrow Decrypt_{sym}\ (C, k_{sym})$
6: end function

During the verification process (Algorithm 4), Verifier generates the hash value $hash_c$ of received data (ciphertext) using the same hash function. Also, Verifier feeds the digital signature and the verification key into the verification algorithm and extract the hash value $hash_p$ of original data (plaintext). If both hash values are equal, it means data file is not modified during transfer between sender and receiver.

Algorithm 4 Signature Verification.

1: Input: Encrypted file C, Signers Public key (sk_{spub})
2: function VERIFICATION (C, sks_{pub})
3:$hash_c$ ← calculate hash of the received encrypted data file C to be verified
4:Using Public key sk_{spub} of signer, extract $hash_p$ of senders file
5:if $hash_c = hash_p$ then
6: return C
7: else
8:return “Signature incorrect”
9: end if
10: end function

Model Implementation

In our system, the patient is equipped with wearable devices such as a blood pressure monitor, insulin pump, or other known devices. Random patients are not allowed to connect with the network, they can only connect once they make an account and provide identity verification documents. Once account verifies the documents provided, users are allowed to access the network. The health information is sent to the smart devices such as a

smartphone or tablet for the formatting and aggregation by the application (see Figure 1). Once complete, the formatted information is sent to the relevant smart contract for full analysis along with the threshold values as required. The threshold value decides whether the health reading is NORMAL as per standard readings or not. If the health reading is ABNORMAL, then the smart contract will create an event and send an alert to the overlay network and to the patient. Also, it stores the abnormal readings to cloud servers and cloud server can then transfer the hash of the stored data to the overlay network. When health data is transferred to the cloud server, the sender adds a digital signature to the data. Overlay network then sends an alert to the health providers. Here, we are not storing the health readings to the overlay network, but we only store the transaction alert to the overlay network.

Health Alert Event should also be anonymous, and privacy preserved to the overlay network. We treat this alert as a transaction of the specific user and apply all advanced cryptographic techniques according to the algorithm explained in Section 6. Here the entity who is sending the information could be treated as a sender and who is receiving the information could be treated as a receiver. Here we only describe the flow of data in our system and do not describe all encryption/decryption technical details as we have already explained the cryptographic techniques in above sections by taking a general model of the sender, receiver, and network. An overlay network contains the public key information of all connected nodes and hash index of the stored data over the cloud. Once the healthcare provider node gets an alert, he/she access the full health reading of patient for which he/she is authorized over the network. We summarize the logical flow execution of the system in Figure 12.

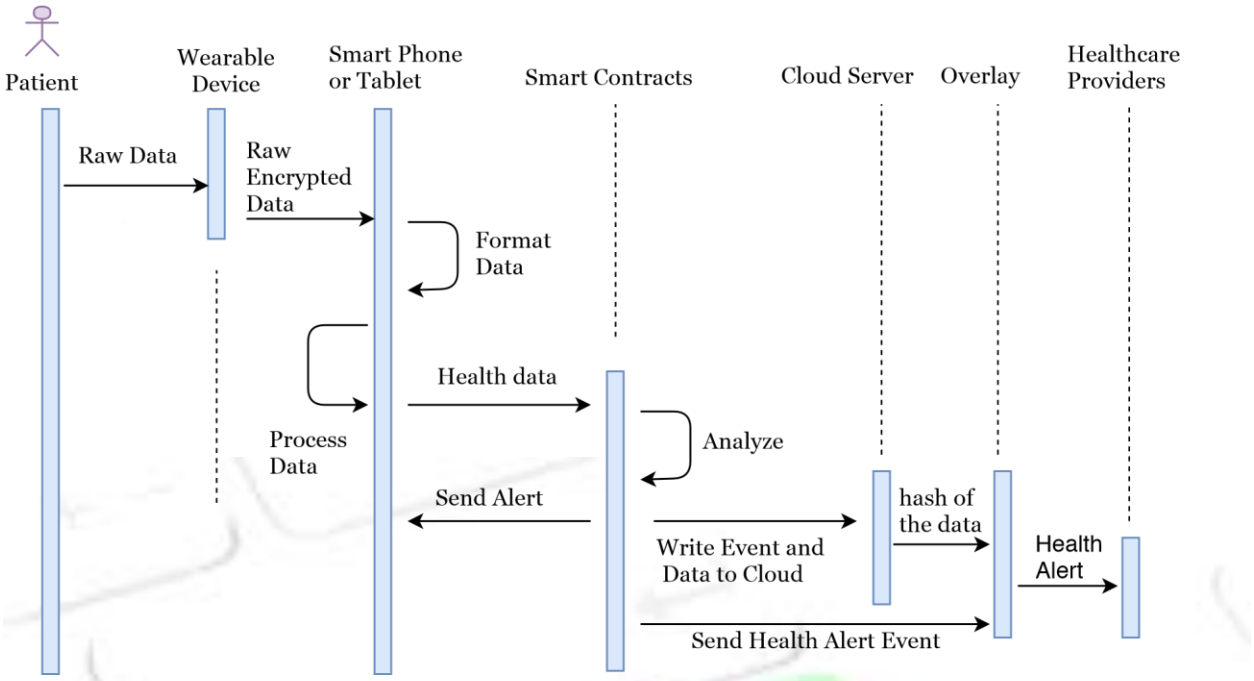


Figure 12. Logical flow execution of the system.

Security Evaluation

In any model, there are three main security requirements that need to be addressed by model designers: Confidentiality, Integrity, and Availability. Confidentiality makes sure that only authorized users can access the system. Integrity is responsible for messages sent to the destination without any change, and Availability means data is always available to the users when needed. We evaluate the security margin of the model under various threats. In this model, the adversary can be a home device, the cluster head, or any other node in the network or part of the cloud storage. These adversaries can discard transactions, sniff communications, create false transactions, or change or delete information from the storage. However, our models' basic aim is to save the network from the adversary, and we focus on this rather than individual nodes. If a node is connected to the network and he verifies proof of authority and registered by the network, then we assume that he is an honest node. In

the case where the network detects some malicious activity by the given node, we can block the malicious node from the network. We summarize the security requirement evaluation in Table 2.

Table 2. Security Requirement Evaluation.

Requirement	Model Solution	Reference
Confidentiality	Proof of Authority, Public Key	Section 6
Authorization	Using Public Key and Lightweight Digital Signature	Section 5.2
User control	Proof of Authority	Section 7
Integrity	Hashing of data blocks	Section 4
Availability	Achieved by limiting acceptable transactions	Section 4
Anonymity	Lightweight Ring Signature	Section 5.3

We considered a few attacks that could be possible in this model and find a security margin against them in our model:

1. Denial of Service (DoS) Attack: In such an attack, the attacker tries to prevent the authentic user from accessing the service in the network. In such cases, the adversary can launch fraudulent transactions and can increase traffic in the network. However, in our system, random users cannot join the network without proof of authority. Regardless, consider an adversary

attacked the network and started sending fake transactions to the network. In such a case, cluster head will check his/her public address and if it is not available or registered with cluster head then it will not broadcast the transaction to the network and forward the request to other clusters. If the public address is not available or is a registered public address, no cluster will accept the request and after many attempts, cluster head will finally block him/her in the network, and therefore our system is safe against this attack. However, a particular attacker can attack many times with a different public address.

2. Mining Attack: Consider an adversary hacked a few cluster heads and started controlling multiple cluster heads. In such a situation, fake mining is possible but once it is detected by other cluster heads or nodes, they can easily trace the fake cluster heads. This is because in our model if any cluster head approves a block then it will add a digital signature over that block and without the digital signature other cluster heads will not accept a new block in the network. Once a fake cluster head is detected by the network it can be modified by the nodes in that cluster.
3. Storage Attack: If an adversary attacked the cloud storage, he/she can remove, change, or add data in the cloud. However, in our model we are using a hash of the data block stored in the cloud, therefore changes in the data can easily be detected. However, in our model, if any user wants to store or manage data over the cloud, he needs to digitally sign the data and can only access his/her data or others' data with permission. In such cases, if someone else modifies the data, the cluster heads can block them from the network.
4. Dropping Attack: For such an attack, the adversary should have to control cluster heads. The cluster heads

under the attacker's control will not be able to do anything in the network. They will drop all received blocks and they will not be connected to other nodes or clusters. As a solution here the nodes in the network can elect another node as a cluster head.

Future Work

This paper takes an initial look at a blockchain-based IoT model glimpsing into an advanced security and privacy model to be used in any current IoT-based remote monitoring system. Our main future direction for this work or any researchers who wish to further this work is to implement the system in a testable system to provide some real work security guarantees apart from what has already been established for all the individual cryptographic components used. We also hope to find an industrial partner to help bring some of the novel ideas mentioned in this work to become commercially available.

Conclusions

IoT privacy and security is one of the most significant issues nowadays in academia and industry. Due to resource constraint factor of IoT, existing security solutions are not well suited. Our proposed architecture provides a solution to most of the security and privacy threats while considering the resource constraint factor of IoT. In this paper, we introduced a novel hybrid approach that combines the advantages of the private key, public key, blockchain and many other lightweight cryptographic primitives to develop a patient-centric access control for electronic medical records, capable of providing security and privacy. We also raise open questions to reduce various attacks such as DoS, modification attacks etc. However, resource-constraints of IoT are key challenges towards answering such problems or seizures.

System Access and Authorization

a. Admin

II. An admin has the privileges to access the entire system and can manage various modules of the system.

1. The admin has the privilege to add and configure new hospitals on the system's chain. Updates on the entire chain can only be authorized by the super Admin.
2. An admin has the privilege to add (register) medical equipment on the system, update and authorize an order and supply of any of the equipment by hospitals on the chain.
3. An admin has the privilege to manage accounting information; Billing and money transactions on the system.
    a. Verification and Authorization of payment modes (Medical covers, online money transactions, mobile money transaction and banking)
    b. Authorization of payments to suppliers
    c. Authorization of payments to staff and other labors
    d. Generation of accounting reports.
4. The admin has the privilege to manage patients;
    a. Viewing and managing clients billing info
5. Verification and authorizing medical records
    a. Verification and authorizing some changes on patient's accounts
6. The Admin has the privilege to staff; Doctors, Nurses, Lab Technicians etc.
    a. Authorization and disbursement of payments to staff
    b. Allocation and delegation of duties to staff
    c. Supervision of some key activities and projects
    d. Authorization of changes made on staff accounts

7. All technical decisions concerning the system to be made by Admins.

b. Doctor

1. A doctor can be an admin hence one will inherit some of the aforementioned privileges.
2. A doctor will have the privilege to view part of the client's/patient's medical record (medical history) that he/she has ever recorded about the patient
3. If a change/update has to be made on the medical record block chain rules will be followed as outlined earlier
4. An access to the client's/Patient's full medical history will only be authorized by the client or Next of Kin where necessary.
5. Note: Full medical record of a client may consolidate data from various hospitals or doctors.
6. A doctor will have access to a drugs database/Pharmacy within a given hospital.
7. A doctor may have access to Nurses information with a privilege to assign a patient to a particular nurse. (Administrative task of delegating duties)
8. A doctor may have access to selected medical research project
9. A doctor will have access to telemedicine conference room and/or theatre.

c. Nurse

1. A nurse will have access to drugs store and/or database within a given hospital
2. A nurse will have the privilege to view the patient's current medical record assigned to him/her.
3. A nurse may be delegated some of the doctor's task to update the patient's current medical record; monitoring patient's health status and updating the record accordingly.

d. Lab Technician.

1. A Lab Technician will have access to a chemical reagents database and/or store
2. The privilege to post and/or update medical diagnostic data to the patient's record and/or systems timeline.
3. Will have access to AlgNos diagnostic tool.
4. Will have access to samples database and/or sample safes.

e. Client/Patient
   1. Will have access to his/her account
   2. Bio information
   3. Medical record
   4. Current medication
   5. Medical/health tips
   6. Billing information
   7. May view a list of doctors and their basic info
   8. May view hospital's basic information

f. Next of Kin
   i. Will have access to patient's data
   ii. May authorize some transactions on the behalf of the patient based on the privileges assigned to him by the patient