applied sciences

MDPI

Article

Homomorphic Encryption Based Privacy-Preservation for IoMT

Mikail Mohammed Salim, Inyeung Kim, Umarov Doniyor, Changhoon Lee and Jong Hyuk Park *

Department of Computer Science and Engineering, Seoul National University of Science and Technology (SeoulTech), Seoul 01811, Korea; mikail@seoultech.ac.kr (M.M.S.); somem123@seoultech.ac.kr (I.K.); mr.danny951008@gmail.com (U.D.); chlee@seoultech.ac.kr (C.L.)

* Correspondence: jhpark1@seoultech.ac.kr; Tel.: +82-2-970-6702

Abstract: Healthcare applications store private user data on cloud servers and perform computation operations that support several patient diagnoses. Growing cyber-attacks on hospital systems result in user data being held at ransom. Furthermore, mathematical operations on data stored in the Cloud are exposed to untrusted external entities that sell private data for financial gain. In this paper, we propose a privacy-preserving scheme using homomorphic encryption to secure medical plaintext data from being accessed by attackers. Secret sharing distributes computations to several virtual nodes on the edge and masks all arithmetic operations, preventing untrusted cloud servers from learning the tasks performed on the encrypted patient data. Virtual edge nodes benefit from cloud computing resources to accomplish computing-intensive mathematical functions and reduce latency in device–edge node data transmission. A comparative analysis with existing studies demonstrates that homomorphically encrypted data stored at the edge preserves data privacy and integrity. Furthermore, secret sharing-based multi-node computation using virtual nodes ensures data confidentiality from untrusted cloud networks.

Keywords: homomorphic encryption; privacy; secret sharing; IoMT

Citation: Salim, M.M.; Kim, I.; Doniyor, U.; Lee, C.; Park, J.H. Homomorphic Encryption Based Privacy-Preservation for IoMT. *Appl. Sci.* **2021**, *11*, 8757. https://doi.org/10.3390/app11188757

Academic Editors: Hassan Chizari, Kamal Bechkoum and Tariq Abdullah

Received: 21 June 2021
Accepted: 17 September 2021
Published: 20 September 2021

1. Introduction

Intelligent healthcare systems in smart cities widely implement Internet of Things (IoT) sensors to monitor patient health. Sensor devices for humans such as smart thermometers record a patient's body temperature [1], Q-bands detect user mobility [2], pacemakers combined with medical alert systems monitor and alert during cardiac arrests [3], and proximity tracers identify potential new contagious disease clusters [4]. Temperature sensors monitor plasma storage in hospitals and are used for future studies [5]. These devices are some of the few medical sensors that monitor a citizen's health. Confidential patient information is ever at risk of being intercepted by attackers attempting to steal personal data and hold hospital and medical research institutions to ransom. Ransomware attacks on hospitals such as Universal Health led to doctors and nurses losing access to essential and critical patient data, requiring them to shift to pen and paper and address immediate concerns [6]. More than 250 hospitals and clinics in the USA were affected, resulting in delayed responses involving critical machine tasks such as blood pressure, oxygen level, and heart rate monitoring [7]. Furthermore, private data processes on public servers are exposed to increasing ransomware attacks, endangering patient healthcare services and risking data loss.

The importance of this study lies in securing the data generated by medical devices and maintaining the privacy of data computations performed by healthcare applications in public servers. Healthcare systems require quick decision-making for high-risk patients suffering from after-effects of medical diagnosis. However, several existing studies rely on performing calculations on the cloud layer due to the vast amount of available computational and storage resources. Latency increases in transmitting results from the Cloud back to devices delays diagnosis for patients where quick decision making is essential for their

health [8,9]. As medical data increases, communication overhead between the Cloud-IoT channel grows due to the frequent and large amount of medical data processing and data exchange in healthcare services [10]. Data analysis at the edge layer, which includes base stations, brings computation closer to the device layer, such as the IoMT reducing latency, bandwidth, and network delay to support real-time applications. Moreover, network bottlenecks and congestions formed during the long transmission path from the device layer to the Cloud layer are avoided using Multi-Access Edge Computing (MAEC) [11].

Several security vulnerabilities are found in the cloud layer, exposing data to intruders via cyber-attacks such as ransomware attacks, resulting in data alteration and manipulation. Furthermore, data analysis in the Cloud requires computation in servers belonging to third-party developers and organizations. Information collected is sold on the dark web to third-party vendors for marketing and advertising organizations. Other malicious users buy data for initiating scams, fraudulent advertisements, and holding data for ransom [12]. Encrypting data using homomorphic encryption prevents attackers from reading private medical data. Moreover, it does not require data decryption at the server-side for data analysis, preventing third-party entities from accessing information shared by Internet of Medical Things (IoMT) devices [13]. Secret sharing is ideal for securely and privately performing computations on untrusted Cloud servers where a user or an intelligent healthcare application does not have control over the privacy of the data [14].

The motivation for writing this paper is to address growing concerns of cyberattacks on healthcare networks. The proposed system resolves two significant problems in IoMT data security: (1) Data integrity and privacy preservation of private IoMT data are maintained when communicating on the network; (2) Computational operations on encrypted data on untrusted cloud servers do not expose patient data.

The main contributions of this paper include:

1. A cluster of virtual nodes is designed around a group of IoMT devices, ensuring untrusted cloud service providers are unaware of computation tasks performed on data.
2. IoT device data are shared using unsecured channels, resulting in delay and replay attacks. Timestamps and nonce values included in packets detect any eavesdroppers on the communication channel. A time synchronization protocol of the IoMT device clock ensures that no malicious node prevents the time reset process.
3. Data transmission from IoMT devices is secured using homomorphic encryption, thus avoiding data manipulation by cyber attackers. Data intercepted by cyber attackers are in an encrypted ciphertext state, preventing any intruder from learning or manipulating the data.
4. Computation of data is performed on the network's edge layer using virtual nodes powered by resources from untrusted cloud services. Offloading operations from the Cloud to the edge prevents untrusted cloud servers from learning the computation process.
5. Using a share-based virtual node selection process, homomorphic secret sharing prevents attackers from learning which selected virtual nodes are used for the computation process.
6. A comparative analysis with existing research shows the proposed scheme provides data confidentiality, integrity, and a privacy-preserving secure computation process.

The remainder of this paper is organized as follows. In Section 2, we discuss technologies for secure encryption of data and the computation process. In Section 3, we present an overview of the proposed system and its workflow process. In Section 4, we perform a comparative analysis of the proposed scheme with recent research studies; weaknesses of other schemes and future research area are also discussed and finally, in Section 5, we conclude our paper.

2. Related Works

In this paper, we implement both homomorphic encryption and secret sharing methods to secure IoMT data and analyze both schemes. We examine both methods, discuss recent related existing researches, and present key considerations for securing IoMT data.

2.1. Existing Research

Homomorphic encryption is based on the principle of processing data without the requirement of decrypting it [15–17]. Data confidentiality is one of the primary principles of implementing the encryption system in untrusted server systems. Compared with other encryption protocols, homomorphic encryption allows algebraic computations such as addition and multiplication directly on encrypted data and treats it as plaintext. This is expressed as,

$$v = f(u) \Leftrightarrow Enc(v) = g(Enc(u)) \tag{1}$$

Here v and u represent unencrypted vectors, while Enc behaves as the encryption operation performed on v and u vectors. We consider a cryptosystem as c with Enc (encryption function), plaintext (p_i), and ciphertext (c_j), where:

$$Enc(p_i) = (c_j) \tag{2}$$

We perform additive (3) and multiplicative (4) operations on p_i such as:

$$\Delta : \ Enc(x_1) \ \Delta \ Enc \ (x_2) = Enc \ (x_1 + x_2) \tag{3}$$

$$\Delta : \ Enc(x_1) \ \Delta \ Enc \ (x_2) = Enc \ (x_1 * x_2) \tag{4}$$

An encryption process satisfying both additive and multiplicative properties is considered homomorphic. A combination of XOR and AND Boolean functions are used in the encryption scheme. The homomorphic encryption $(H(Enc))$ scheme is further divided into four algorithms, which are as follows:

$$H(Enc) = (key\ generate,\ encrypt,\ decrypt,\ evaluate) \tag{5}$$

- Key generation is based on security parameters (σ) taken as input, and the algorithm outputs an encryption (k_e)/decryption (k_d) key pair.
- The encryption algorithm takes input $(p_1,\ p_2, \ldots\ldots, p_i)$ and is divided into bits $(b_1,\ b_2, \ldots\ldots b_i)$ of 0 and 1s and ciphertext $(c_1,\ c_2, \ldots\ldots c_j)$.
- The decryption algorithm reveals data by receiving input p_i to provide output $c_1,\ c_2, \ldots\ldots c_j$.
- Evaluation of the algorithm takes input p_i with function (f) and reveals output $c_1,\ c_2, \ldots\ldots c_j$.

Homomorphic encryption has three types of encryption schemes: partially homomorphic, somewhat homomorphic, and fully homomorphic encryption [18].

- Partially homomorphic: A single type of mathematical operation is permissible on $Enc(p_i)$—either addition or multiplication performed without any limitation in execution.
- Somewhat homomorphic: There is a restriction on the execution limit, with only a certain number of times that either addition or multiplication are permissible.
- Fully homomorphic: Enables a larger subset of mathematical operations on the p_i without any limit on the number of executions. There is a limitation on the number of users it supports; its implementation results in a significant computation overhead and increases network latency due to slow runtimes.

Shamir [19] proposed the concept of secret sharing [20] by dividing data (x) into multiple and reconstructible different pieces (k), where no piece of information $(k-1)$ reveals d. We assume that the secret data d is confidential information, and break d into an n number of pieces $(d_1,\ d_2, \ldots\ldots,\ d_n)$ in a way that satisfies two conditions:

- A third entity requires a certain pre-fixed number of k pieces of $d_1,\ d_2, \ldots\ldots,\ d_n$ to make d accessible.
- The possession of $k-1$ makes d unaccessible, with the condition that each element of d_i is by itself insufficient to reveal the message.

The determination of selection of $d_1, d_2, \ldots\ldots, d_n$ is known as the threshold value, and is especially helpful in securing keys used to encrypt plaintexts. The reconstruction of the key (k) requires all threshold members to announce their individual shares to reconstruct k and gain access to the secret d.

Recent research for securing IoMT data focuses on encrypting using homomorphic encryption for ensuring data privacy. Lin et al. [21] presented a new privacy-enhanced data fusion strategy for encrypting data using homomorphic encryption and suggested an incentive-based approach to encourage users to share data with the healthcare network. The process combines mobile edge computing (M.E.C.) and IoMT, where various M.E.C. servers support COVID-19 applications, store, process, and analyze results. In place of using Cloud layer resources, all data computation is performed at the edge layer. Vizitsu et al. [22] proposed a fully homomorphic encryption scheme to perform neural network operations directly on floating-point data with reduced computational overhead. A vulnerability in the proposed Matrix Operation for Randomization or Encryption-based homomorphic encryption system is that the key is vulnerable to decryption. An attacker using an optimization problem with access to a large set of key pairs can determine the encryption key, resulting in data security and confidentiality issues. Raw data is uploaded to an external and private server on the Cloud for security, but the proposed scheme fails to protect it in transmission. Cheng et al. [23] presented a patient privacy protection model based on federated learning verification. The model combines homomorphic encryption, blockchain, and federated learning to ensure the privacy of medical records as data learning is performed both centrally and locally. Each client/user/device trains the model using their local dataset collected and transmitted to the central server. Yang et al. [24] proposed a framework for medical image security. Firstly, the authors present a reversible data hiding scheme for image quality improvement and the embedding of privacy data in images. Secondly, medical image encryption is managed using a homomorphic encryption method using the chaotic map.

Other research uses and relies on different encryption standards, such as the Elliptic Curve Digital Signature Algorithm (ECDSA) and technologies such as blockchains, to secure IoMT data. Cano et al. [25] addressed the security and privacy concerns of IoMT data using the ECDSA to generate dual digital signatures. Data transmission from devices to the cloud layer via the edge is authenticated from valid sources using dual signatures. Signature verification is performed using edge layer resources due to their lack of resource constraints and agility compared to IoMT devices. The research objective is to prevent the edge from accessing the encrypted data and the Cloud from learning the user's identity. Data decrypted at the Cloud is exposed to external servers accessing data and other malicious cyber-attacks. Nguyen et al. [26] proposed a decentralized architecture combining data sharing and offloading methods; the computation of data is performed at the M.E.C. for increased speed and user privacy. The novelty of this architecture lies in implementing smart contracts for authentication and traceability during data sharing. A central authority or entity requirement for approving a user access request to upload data is removed using smart contracts that establish the user's validity. Data integrity is maintained as information is stored in blocks, and any tampering will be reflected in modified hash values. Data lookup time is reduced due to the direct storing of hash values in smart contracts, providing direct data access in the Interplanetary File System.

2.2. Key Considerations

In order to maintain a secure and privacy-preserving edge–cloud ecosystem for IoMT, the following key considerations are essential to our proposed scheme,

- Data Confidentiality: Communication of data from IoMT devices to the edge layer transfers the ownership and control of the data. The physical security offered by the devices is lost when transmitted to external network layers, such as the base station at the edge layer. Data confidentiality schemes are required to ensure that any unauthorized entity viewing the data does not observe it in plaintext format.

Information is required to be encrypted, preventing any form of identification of a user or device. Devices or applications supporting healthcare systems are needed to encrypt the data to a ciphertext, allowing data sharing on untrusted edge and cloud servers.

- Data Integrity: Data transferred from IoMT devices to insecure edge servers are vulnerable to foreign intrusion attempts, resulting in data manipulation and affecting the reliability of the final patient data computation results. It is essential to maintain data integrity during transmission and computation.
- Privacy-Preservation: Privacy is a significant challenge when transmitting sensitive and private user data using untrustworthy external networks such as cloud servers and edge nodes. Third-party edge and cloud service providers can learn much sensitive information about personal user data and sell it for profitable gains. Data sold to other entities allow fraudulent, targeted marketing campaigns and scams, resulting in further stress and financial loss to users. Encryption of patient identity and medical diagnosis information is essential using pseudo-random permutation and public-key encryption techniques.
- Secure Data Computation: Data shared across edge entities are often owned by external service providers that require information to be decrypted before performing mathematical operations. Encryption techniques such as homomorphic encryption allow edge service providers to perform computation operations without requiring the user to decrypt the data.

3. Proposed Scheme

This section presents an overview of the healthcare environment and the components used in the proposed scheme to secure IoMT data. The environment consists of a hospital that houses patients and several IoMT devices that transmit data to the physical base station at the edge layer. Data at the edge is homomorphically encrypted, and the intelligent healthcare application sends computation queries for analysis on collected patient data. Base stations are supported by cloud computing resources for extensive computational operations and assign several virtual edge nodes with equal resources around the hospital. Virtual edge nodes are selected by the base station using secret sharing, which prevents untrusted cloud servers from learning the performed data computations. Furthermore, in the workflow subsection, we describe the particular method implemented to design virtual nodes at the edge layer, secure IoMT–edge data communication, encrypt data, and secure and preserve computation processes.

3.1. Overview of the Proposed Scheme

The proposed scheme consists of several elements participating in secure encryption, data management, and privacy-preserving computation processes:

- IoMT devices identify as the data owners (DO) responsible for generating data using embedded sensors in each patient. These machines include medical devices embedded in beds, machines deployed for scanning a patient's vitals, and other smart sensors surrounded in the hospital.
- Virtual edge nodes (VEN) collect encrypted data shared by IoMT devices and use them to process and transmit output results to the healthcare application.
- The intelligent healthcare application, behaving as the data user (DU), monitors the data shared by IoMT devices. The DU manages the homomorphic decryption process and the number of shares generated during the homomorphic secret sharing computation process.

As shown in Figure 1, the proposed scheme is based on the following processes:

Step 1. Several varying IoMT devices present in the hospital collect data from patients and transmit it to the DU that manages the intelligent healthcare application.

Step 2. The DU forwards the data to the base station, which encrypts the plaintext data (x) to produce an encrypted ciphertext (c) using homomorphic encryption. Here, $c = Enc(x)$ represents data being encrypted.

Step 3. The edge-based base station provides several VENs around hospital zones for various computation services assigned by the DU for different tasks. A cluster is designed that provides support to IoMT devices that behave as cluster members along with VENs.

Step 4. A cluster head (CH) behaves as the centroid of each cluster, and the one closest to the physical set of IoMT devices is selected. CH enables a line of sight with cluster node members based on distance proximity, providing better connectivity. In our paper, the base station maintains the geographical coordinates of each device and assigns a selected VEN as the CH closest to the set of devices using k-nearest neighbors.

Step 5. Data transmission from the IoMT device is sent to the base station using unsecured channels, which exposes them to replay and delay attacks. Timestamp and nonce value-based scheme alerts the device to an attacker present in the communication channel.

Step 6. Time clocks on devices are subject to being reset by attackers, and a time resynchronization scheme on the VENs and the cluster head VEN securely resets the time clock.

Step 7. Data is encrypted from plaintext to ciphertext using a homomorphic encryption scheme at the base station and prevents cyber attackers from accessing the encrypted data.

Step 8. Homomorphic secret sharing schemes rely on the number of computational operation queries requested by the DU to generate an n number of shares ($Sh_1,\ Sh_1, \ldots\ldots Sh_n$) distributed to each cluster head VEN. Data received at each node is previously encrypted using the homomorphic encryption algorithm.

Step 9. Each cluster head VEN evaluates and computes data using cloud service resources allocated by the base station. Allocation of resources is equally distributed to each VEN, preventing an untrusted cloud service from learning the computation operations.

Step 10. The DU receives the output data from the various cluster member VENs via the base station and performs the final decryption process of converting the results from the VENs ciphertext to plaintext.

IoMT devices include sensors with low-powered battery and computational resources, such as Bluetooth Low Energy and Low Power Wide Area Network devices that continually transmit data to the DU. Data is encrypted at the base station using homomorphic encryption, which supports addition and multiplicative operations. DU sends queries to the base station to perform computational tasks on the encrypted data. Heavy operational tasks are difficult for a single edge node to function due to limited computational power. Therefore, the proposed scheme implements the concept of the multi-party computation (MPC) model, using multiple VENs to distribute the task of computation and ensure computation privacy.

The selection and deployment of different VENs are based on the physical location of IoMT devices; a hospital has hundreds of patients, each using multiple sensors located in varying geographical areas within the building. Physical edge nodes are avoided for data calculation operations due to their lack of resources for managing the big data generated from healthcare applications. Different cloud services are used to acquire resources and dynamically provide resources to the DU. Selection of varying cloud services prevents a single untrustworthy third-party service provider from storing performed additional and multiplicative computational operations and learning the data analysis process. Compared to cloud services, virtual edge nodes performing computation operations on homomorphically encrypted data provide results closer to the devices, resulting in reduced latency in transmitting time-sensitive output to the DU.

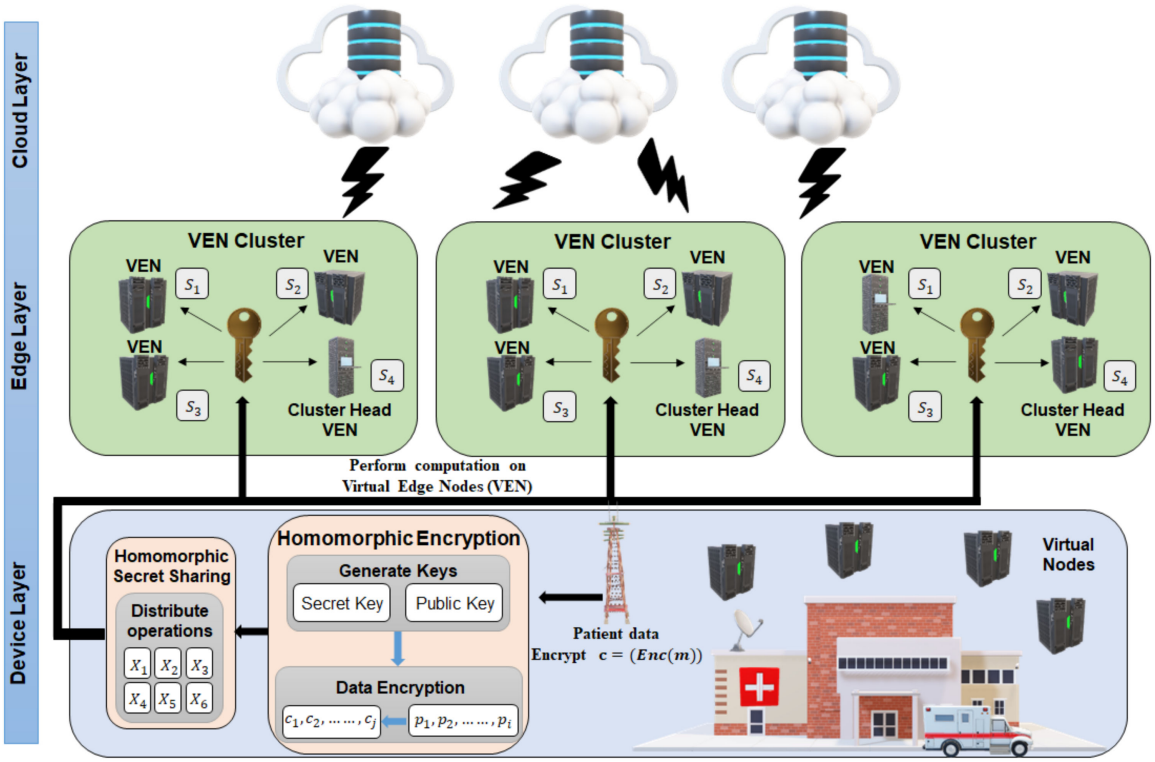

Figure 1. Proposed scheme overview.

VENs benefit the DU by providing dynamic allocation of nodes and serving as resource provisioning services. A single or selection of nearby physical edge nodes, such as base stations, cannot offer real-time services due to limited inbuilt computational resources. This paper assumes that the healthcare service deploys cloud-based resource services such as Amazon Web Services and Google Cloud, using on-demand payment services. On-demand payment-based services provide a highly scalable selection of edge nodes, ensuring complex operations are performed without delay.

3.2. Workflow of the Proposed Scheme

This section describes the process flow of the proposed scheme based on four phases: First, clusters are assigned for a set of virtual edge nodes (VEN) and IoMT devices. Secondly, data transmission from IoMT devices to the physical base station is secured to identify any possibility of replay and delay attacks. Thirdly, data received by the base station is encrypted using homomorphic encryption. Fourthly, secret sharing distributes encrypted data to multiple VENs in a cluster, preventing untrusted cloud servers from determining the computation process. Finally, the decrypted results by the DU are sent back to the IoMT devices (DU). Figure 2 illustrates the workflow process of all four phases in the proposed scheme.

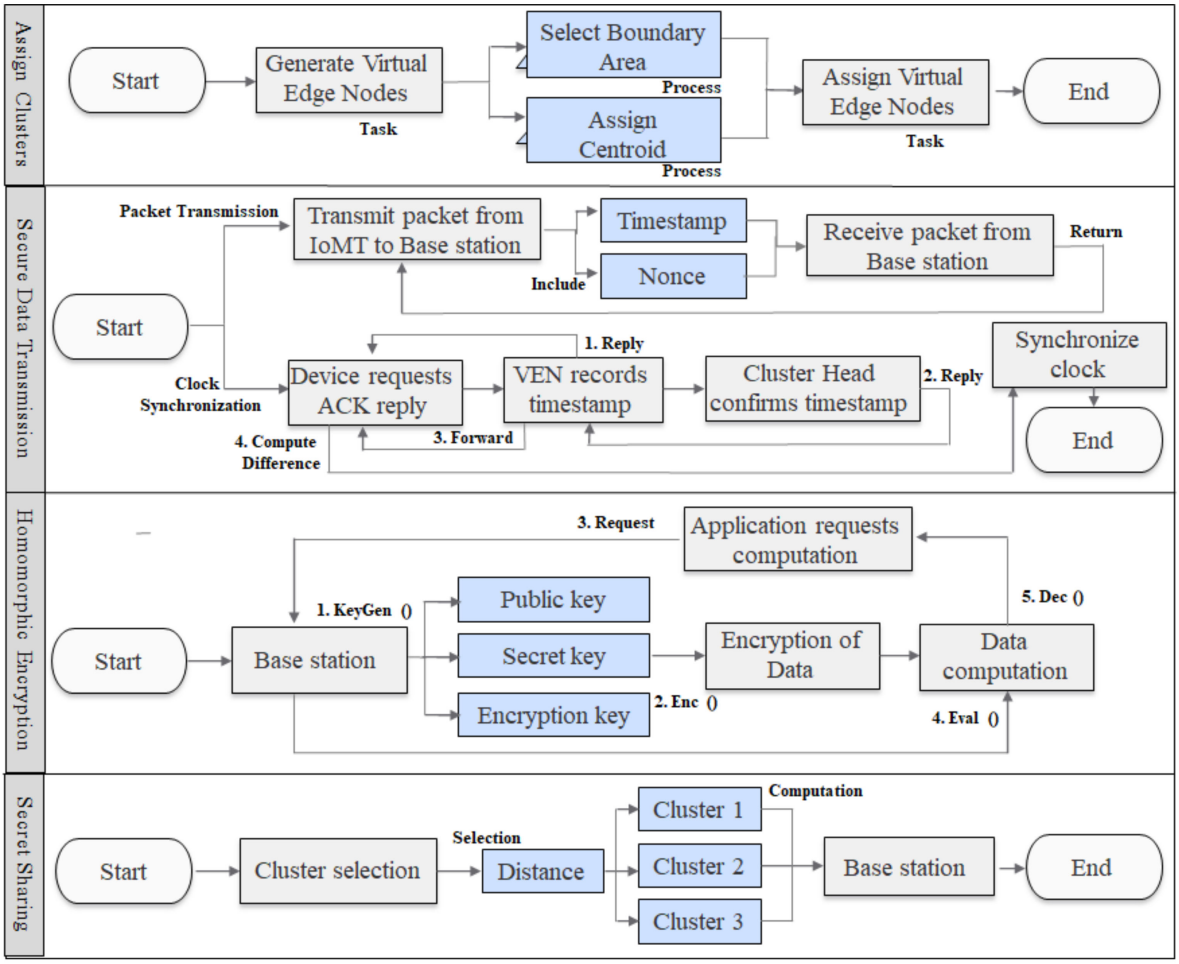

Figure 2. Workflow of the proposed scheme.

3.2.1. Assigning Clusters

We assume that the local physical edge nodes, such as a base station, rely on the vast computing capabilities of the expansive cloud computing environment. Physical edge nodes have limited resources to compute extensive data shared by IoMT devices in the healthcare network. The objective is to design several virtual nodes with equal computing for data computation using the homomorphic secret sharing scheme. The process of cluster generation is explained as follows:

Step 1. Initially, all VENs are generated and dispersed randomly to cover the entire coverage area of a hospital. Each VEN contains and stores the distance from the virtual node to the central physical base station. Each VEN has equivalently distributed computing resources received from the cloud environment services.

Step 2. Several clusters (C_n) are designed for each group of IoMT devices and their respective VENs. Each cluster includes a boundary area that consists of selected VENs and sensors using k-nearest neighbors (KNN) based on a preset distance. The objective is to prevent any two clusters from including the same sensor device in their group.

Step 3. Geolocation data of each device and their distance from the central base station is used to include it in the cluster. Each cluster has a unique identity $(C_1, C_2, \ldots\ldots C_n)$ and each device registered with the cluster cannot join another cluster.

Step 4. Communication overhead is a problem when dividing devices into multiple clusters. Each device may transmit data using VENs with a longer distance from the physical base station than other VENs. A centroid (*cen*) node is selected as a

midpoint in the cluster using geolocation data and ensure each device $(d_1, d_{2,\ldots,}d_i)$ is equidistant from the centroid. Each centroid is a part of the set of VENs $(V_1, V_{2,\ldots,}V_i)$ included in each cluster and is represented as $VEN \in (V_1, V_{2,\ldots,}V_i)$. The centroid VEN is selected as the cluster head and provides an equivalent data communication distance for each IoMT device in the cluster.

3.2.2. Secure Data Transmission

Performing secure encryption and generating keys directly on computationally challenged IoMT sensor devices is challenging. Replay attacks and delay attacks are a constant threat when sending unsecured plaintext data across the network. An attacker reading a packet and sending it at a later time is termed a replay attack. In contrast, delay Attacks are caused by an attacker intercepting the packet, suspending sending it to its destination address, and sending it at a later time. Timestamps in combination with nonce value are used to prevent attackers from replay and delay attacks.

The IoT device sends timestamps and nonce values as a part of the packet header, using the MAC address header to the physical base station. The base station accepts packets from the device based on a time tolerance and sends a reply to the device of the packet being received on time. However, an attacker can send the packet repeatedly at a later time as a flood attack to negatively affect the base station's performance. Including the nonce value with the timestamp secures the base station by dropping all delayed packets using the same nonce value as the previous packet.

A successful attack results in an IoT device not using the same network channel to communicate with the base station. However, this method is not secure. The attacker resets the timestamp clock on the sensor and forces it to send future packets with incorrect parameters, causing all packets to be dropped. Qiu et al. [27] proposed a secure time synchronization protocol mechanism for IoT devices by coordinating the clock with the father and grandfather nodes. Other device nodes are ignored as they are also vulnerable to being malicious nodes. A spanning tree-based protocol protects the device from fake timestamps. The proposed scheme includes the protocol as shown in Figure 3, ensures the IoMT device clock is reset to the correct time, and the process is as follows:

Step 1. The device communicates with nodes that are part of its local cluster to correctly reset the time and ignores other devices as they can provide incorrect timestamps.

Step 2. The device sends a message to a VEN included in the cluster and requests for an ACK reply. The selection of VEN to communicate with is based on the distance to the physical device.

Step 3. The VEN records both the timestamps of when the packet was sent by the device, recorded as $tsmp_1$ and when the message is received as $tsmp_2$. The second timestamp indicates that the message received is within the acceptable threshold value.

Step 4. The ACK reply message sent back by the VEN to the device includes $tsmp_3$ which indicates when the VEN sent the ACK message. The device records the received ACK message as $tsmp_4$.

Step 5. The clock difference calculated by the device measures $tsmp_1$, $tsmp_2$, $tsmp_3$, and $tsmp_4$, received during its communication with the VEN. The difference is calculated as:

$$\Delta\ (d_i,\ v_i) = \frac{(tsmp_2 - tsmp_1) - (tsmp_4 - tsmp_3)}{2}$$

Step 6. The VEN further sends a message to the cluster head node for time confirmation and requests an ACK message as a reply.

Step 7. The cluster head node receives the message and records $tsmp_6$. The ACK reply message sent back to the VEN now includes both $tsmp_6$ and $tsmp_7$.

Step 8. The message is forwarded at the received time to the device. The device records the received message's $tsmp_8$.

Step 9. The device now has further knowledge of the clock time difference between the cluster head and the device based on the received $tsmp_1$, $tsmp_2$, $tsmp_3$, ..., $tsmp_8$. The clock difference is measured as follows:

$$\Delta\ (d_i,\ cen) = \frac{(tsmp_6 - tsmp_1) - (tsmp_8 - tsmp_7)}{2}$$

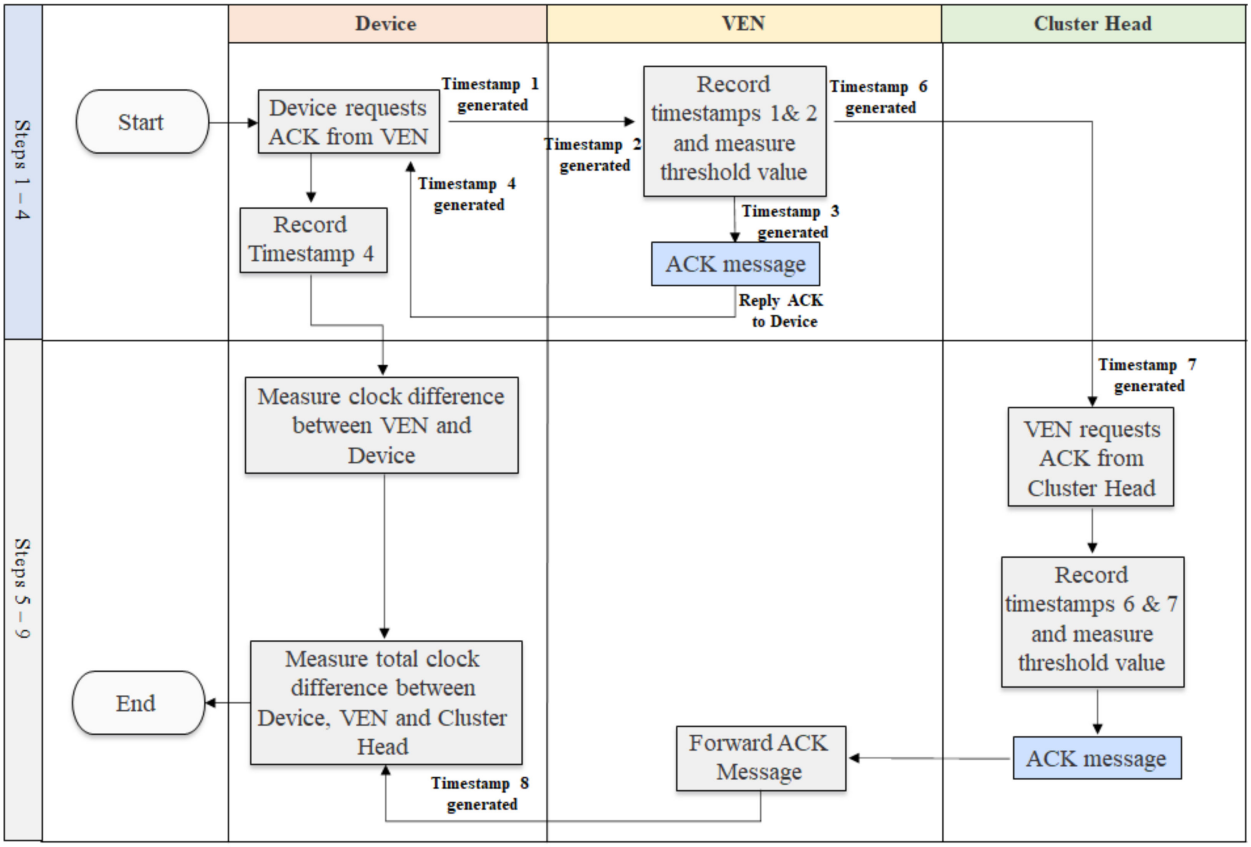

Figure 3. Time synchronization protocol.

For additional security, the acceptable threshold value margin of error is calculated based on a single hop between the device, VEN, and cluster head. If multiple hops are detected during transmission between the device and the VEN, the device identifies the VEN as a malicious node; the time synchronization is instead performed with the cluster head.

3.2.3. Homomorphic Encryption

The proposed scheme's data privacy features are ensured using homomorphic encryption of the plaintext data received from individual IoMT devices. The entire encryption process is performed at the local base station. The objective in this phase is to ensure that individual data (x_i) received from $(d_1,\ d_2 \ldots\ldots, d_i)$ is not available as plaintext $(p_1,\ p_2, \ldots\ldots\ p_n)$ for cyberattackers and untrusted cloud servers. To solve this, we encrypt the data using $c_i = Enc(x_i)$, where (c) represents the ciphertext data and is used to perform further computation operations.

Homomorphic encryption includes four algorithms, which are as follows:

1. KeyGen (): The Key Generation algorithm uses a KeyGen () function to construct three keys, the public key (pk), secret key (sk), and a verification key (vk).
2. Enc (): The encryption algorithm *generates* the ciphertext (c) using parameters including the sk, the dataset (x), and the label (l) of the dataset. The Enc () function is represented as $Enc\ (sk,\ l,\ x)$.

3. Eval (): The evaluation algorithm at the physical base station receives a query from the intelligent healthcare application (DU) to perform computational operations. The Eval () function includes parameters, pk, the homomorphic function (f), the ciphertext (c) and is represented as $Eval\ (pk,\ f,\ c)$.
4. Dec (): The DU using the decryption algorithm verifies the computed data received from the base station and chooses to accept or reject it. The Dec () function is represented as $Dec\ (sk,\ DUQ\ ,\ c)$, where DUQ is the query sent by the DU.

The base station creates the pk, sk, and encryption keys (k_e) using KeyGen (). Data encryption initiates with inputs sk, $l \in \mathcal{L}$, and $x \in$ for Enc (). The resultant output of the function is c.

Computation of encrypted data using the Eval () function is performed to derive powerful analytics for the DU and the DO. Such heavy tasks are performed by computationally powerful machines which do not include the IoMT devices. Mathematical operations, such as addition and multiplication, completed on the cloud environment directly result in a lack of computation privacy. A third-party cloud service is an untrusted entity, and a cyberattack on the cloud layer exposes the operations performed on the network. This paper assumes the healthcare network implements a cloud service to provide computational resources to the local physical base station. The homomorphic Enc () function performs encryption directly on the base station, and the computation operations are performed on the VENs. The VENs preserve computational privacy, as each node is provided with an equivalent amount of resources to perform computation operations. In cloud-based systems, third-party service providers attempt to guess the data output by observing the addition and multiplicative functions executed, resulting in exposure of private user data on the network.

The DU requiring the output of the selected mathematical computation operation on the data selects the k_e and the ciphertext $c_1,\ c_2,\ \ldots\ldots,\ c_j$ and provides them as input for $Eval\ (k_e,\ op,\ c_1,\ c_2,\ \ldots\ldots,\ c_j)$. Data evaluation requires it to be shared with different VENs using the homomorphic secret sharing mechanism. The final computation result is a ciphertext transmitted to the DU, decrypting the ciphertext using the Dec () function.

3.2.4. Homomorphic Secret Sharing

This paper proposes that multi-party computation is performed at separate VENs to hide operations from the untrusted cloud network. We implement a homomorphic secret sharing (HSS) scheme to divide computation functions across different VENs. Each share's data is encrypted using the Enc () function described in the homomorphic encryption process. The process flow of secret sharing is described in the following steps:

Step 1. The base station encrypts the data using the ENC () function as part of the homomorphic encryption process.
Step 2. Different data bits collected from varying IoMT devices (DO) have separate computation processes required by the DU. The encrypted ciphertext, using Enc (), is forwarded to each cluster head VEN and other nodes in different clusters.
Step 3. The selection of clusters is based on the distance measured from the physical base station. As the number of computation tasks grows, other cluster heads are selected based on their distance proximity with the previous cluster head. If the resources required for computation is high, then the computation is shared with other virtual nodes present within the same cluster.
Step 4. The threshold value for share (sh_n) generation is based on the computation requests generated by the DU where $DUQ_n = (sh_1,\ sh_2,\ sh_3, \ldots\ldots,\ sh_n)$.
Step 5. The base station stores the assigned cluster heads assigned as shared to perform the computation tasks.
Step 6. Data computation results are transmitted back to the base station, which is aware of each assigned node required to return computed data. Data received from other VENs are rejected as unauthorized packets.

Finally, DU performs the DEC () function on the results to obtain plaintext. DU forwards the final data to the DO IoMT devices.

4. Discussion

In this section, we analyze the security of the proposed scheme, and discuss comparison results with other existing research. The four key areas of consideration, data confidentiality, integrity, privacy-preservation, and secure data computation, are essential to secure and preserve the privacy of IoMT data.

We observe from the summary in Table 1 that existing research either do not or only partially satisfy the key areas of consideration for a complete privacy-preserving scheme for securing IoMT data in a healthcare system. Implementation of homomorphic encryption [21–24] ensures data privacy stored in local edge nodes and cloud environments. Lin et al. [21] performed a hypothesis test to ascertain the reliability of collected data and encrypts each set of grouped information for user privacy at the data fusion center. Their research is focused on ensuring the reliability of data and achieving privacy of patients using homomorphic encryption. Computations of encrypted data are performed at local physical datacenters where mathematical operations are exposed to external entities. The vital challenge of masking operations from untrusted entities is not addressed. Vizitiu et al. [22] used keys to encrypt data using linear functions. However, an attacker exposes the keys using an extensive database of values associated with encrypted keys. Evaluation results show that the ciphertext data used for computation is exposed as plaintext using an optimization problem.

Similarly, in [23–26], the objective of the research and evaluation results focuses on encrypting data for privacy but does not provide masking for computation operations from untrusted service providers such as cloud operators, IoT devices and mobile edge datacenters. Our proposed scheme benefits from secret nodes at the local edge layer, preventing untrusted cloud providers from learning the computation operations. Keys generated using secret sharing prevents an attacker from reconstructing distributed data among several VENs and learning the final key used to create shares. Furthermore, existing research does not address the lack of a secure communication channel between the device and base station. Data reliability in our scheme is preserved and ensured using the timestamp-based clock synchronization method. The advantage of the proposed scheme compared with existing research is based on its complete security of data, starting from when devices share it with the local base station. Data storage and integrity are maintained using homomorphic encryption, and finally, operation privacy and security are ensured using secret sharing and VENs.

Medical data requires constant computation using various artificial intelligence methods to analyze and provide output to intelligent healthcare systems. Exposure of computation operations exposes data to untrusted cloud servers, allowing possible deciphering of the type of data stored in Cloud. Third-party cloud services often rely on selling user data to further untrusted parties that use it for their marketing operations or to promote fraudulent schemes and advertisements. Schemes performing computation operations on local IoT devices based on a distributed manner rely on sensor devices with weak inbuilt security protocols. Cyberattacks such as botnet attacks expose operations to malicious parties. It is essential to perform operations directly on network nodes that are maintained by healthcare services and are closer to the device layer for reduced latency. The proposed scheme performs operations on the edge layer using VENs. In this paper, we assume that the healthcare network consumes cloud services for increased computational power at the edge layer. Multiple VENs are designed surrounding the hospital and are distributed with an equal set of computational power to prevent a cloud service from determining which node was used to perform mathematical operations. Furthermore, secret sharing prevents an attacker or the cloud service from determining which VEN cluster was selected for computation operations.

Table 1. Comparative analysis of the proposed scheme with related research.

References	Mechanism	Data Confidentiality	Data Integrity	Privacy-Preservation	Secure Data Computation
Lin et al. [21] (2020)	Homomorphic encryption and mobile edge computing.	Ciphertext output is secured using homomorphic encryption.	Encrypted medical data prevents attackers from manipulation.	Privacy of ciphertext is preserved, however, operations are exposed on public nodes.	Data computation takes place at untrusted edge nodes.
Vizitiu et al. [22] (2020)	Homomorphic encryption	Encryption protects data from being exposed to malicious entities.	Weak encryption keys are vulnerable using an optimization problem.	Exposed keys enable attackers to decrypt data and acquire private medical data.	Arithmetic operations are performed over an untrusted cloud server exposing computation process.
Cheng et al. [23] (2020)	Homomorphic encryption, blockchain, federated learning	Homomorphic encrypted data and local computation using federated learning protects device data.	Encryption prevents attackers modifying data without decryption.	Federated learning preserves privacy by computing data locally on IoT devices.	Weak IoT device security and vulnerability to cyberattacks compromises future computation operation capability.
Yang et al. [24] (2019)	Homomorphic encryption using chaotic mapping	Medical images are encrypted using homomorphic encryption.	Encrypted image data is safe from malicious users.	Image data is only accessible to healthcare services.	The research does not identify any means of preserving computational privacy and only addresses encrypting data.
Cano et al. [25] (2020)	Dual signatures using Elliptic Curve Digital Signature Algorithm	The dual signature scheme does not encrypt medical data but only the identities of patients.	Data is exposed both on the Cloud and the edge, risking manipulation.	Privacy of users is maintained by verifying identities at both the edge and the cloud layer using dual signatures and identifiers.	Any computation of data on untrusted servers requires prior decryption. Data computational privacy is not preserved.
Nguyen et al. [26] (2021)	Blockchain and smart contracts	Data is exposed on public blockchain networks.	Data stored in blocks prevents manipulation as it would alter the hash value.	Privacy of user data is compromised at public blockchain-based networks. Mobile edge computing-based computation exposes mathematical operations performed on data.	Computational operations are exposed at the mobile edge computing layer. Data is required to be decrypted before any operation is performed.
Proposed scheme	Homomorphic encryption and secret sharing	Data is encrypted using homomorphic encryption maintaining confidentiality of collected private medical data.	Data manipulation is not possible on encrypted data, preventing attackers from reading and altering data.	Data privacy is maintained as it is in the form of a ciphertext, preserving both medical data and user identity.	Computations are performed using secret sharing on nodes at the edge layer. Mathematical operations are distributed and performed on hidden nodes.

Other research literature uses ECDSA [25] and blockchain-based decentralized networks [26] to secure user identities, but fail to secure the data from being exposed to untrusted parties. Blockchains provide data integrity, and dual signatures using ECDSA authorize edge nodes and cloud services but require all encrypted data to be decrypted and then analyzed. Plaintext data exposes both user and data privacy at both the Cloud and the edge layer. The proposed scheme relies on homomorphic encryption and secret sharing on VENs to protect both user and data privacy without any requirement of decryption of data for further analysis.

Data shared in the proposed scheme from IoMT devices to edge nodes are performed over insecure channels. The deployment of nonce values and timestamps in packets sent

and received between the edge and device layer alert the network of possible relay and delay attacks. However, there is a need for a more secure and trusted data communication channel between the device and the physical base station. In future research, we intend to focus on securing device–edge data transmission using a lightweight encryption scheme.

5. Conclusions

This paper presented a privacy-preserving scheme for data generated by IoMT devices and stored for further encryption and computation using homomorphic encryption and homomorphic secret sharing. Data transmitted from IoT devices to the base station are protected using a clock synchronization scheme. Timestamps and nonce values prevent replay and delay attacks on the network. Cloud services support edge-based base stations and virtual nodes with extensive computational resources for secure and near device encryption and data computation. Virtual edge nodes operate as multi-party computation nodes combined with homomorphic secret sharing that prevent untrusted cloud services from eavesdropping on circuit operations. A detailed comparative analysis between the proposed scheme and recent research studies are discussed. Future research focus is to secure IoT–edge data communication using a lightweight encryption mechanism.

Author Contributions: Conceptualization and methodology, M.M.S.; software, M.M.S. and U.D.; formal analysis and investigation, M.M.S. and I.K.; writing—original draft preparation, M.M.S., I.K. and U.D.; writing—review and editing, J.H.P., C.L. and M.M.S.; supervision, J.H.P.; funding acquisition, J.H.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Energy Cloud R&D Program (2019M3F2A1073386) through the NRF (National Research Foundation of Korea), both funded by the Ministry of Science and ICT.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Javed, A.R.; Sarwar, M.U.; Beg, M.O.; Asim, M.; Baker, T.; Tawfik, H. A collaborative healthcare framework for shared healthcare plan with ambient intelligence. *Hum.-Cent. Comput. Inf. Sci.* **2020**, *10*, 1–21. [CrossRef]
2. Singh, V.K.; Chandna, H.; Kumar, A.; Kumar, S.; Upadhyay, N.; Utkarsh, K. IoT-Q-Band: A low cost internet of things based wearable band to detect and track absconding COVID-19 quarantine subjects. *EAI Endorsed Trans. Internet Things* **2020**, *6*, 4. [CrossRef]
3. Caring Home, 3 Best Medical Alerts Systems for Those with Pacemakers. Available online: https://www.caring.com/best-medical-alert-systems/best-medical-alert-systems-for-those-with-pacemakers/ (accessed on 6 September 2021).
4. Triax Technologies, Contact Tracing IoT Solution. Available online: https://directory.newequipment.com/classified/contact-tracing-iot-solution-253439.html (accessed on 2 February 2021).
5. COVID-19 Vaccine: The Role of IoT. Available online: https://www.iotforall.com/the-role-of-iot-for-the-covid-19-vaccine (accessed on 2 February 2019).
6. Cyberattacks Cost Hospitals Millions during COVID-19. Available online: https://www.wsj.com/articles/cyberattacks-cost-hospitals-millions-during-covid-19-11614346713 (accessed on 2 February 2021).
7. As Hospitals Cope with a COVID-19 Surge, Cyber Threats Loom. Available online: https://apnews.com/article/us-news-vermont-coronavirus-pandemic-burlington-hacking-28af71f3861d245df052f06e12475c2d (accessed on 2 February 2021).
8. Daoud, W.B.; Obaidat, M.S.; Meddeb-Makhlouf, A.; Zarai, F.; Hsiao, K.F. TACRM: Trust access control and resource management mechanism in fog computing. *Hum.-Cent. Comput. Inf. Sci.* **2019**, *9*, 1–18. [CrossRef]
9. Megouache, L.; Zitouni, A.; Djoudi, M. Ensuring user authentication and data integrity in multi-cloud environment. *Hum.-Cent. Comput. Inf. Sci.* **2020**, *10*, 1–20. [CrossRef]
10. Yang, J. Low-latency cloud-fog network architecture and its load balancing strategy for medical big data. *J. Ambient Intell. Humaniz. Comput.* **2020**, 1–10. [CrossRef]
11. Pham, Q.V.; Fang, F.; Ha, V.N.; Piran, M.J.; Le, M.; Le, L.B.; Hwang, W.J.; Ding, Z. A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art. *IEEE Access* **2020**, *8*, 116974–117017. [CrossRef]
12. Mazi, H.; Arsene, F.N.; Dissanayaka, A.M. The influence of black market activities through dark web on the economy: A survey. In Proceedings of the Midwest Instruction and Computing Symposium, Milwaukee, WI, USA, 3–4 April 2020.
13. Salavi, R.R.; Math, M.M.; Kulkarni, U.P. A Survey of Various Cryptographic Techniques: From Traditional Cryptography to Fully Homomorphic Encryption. In *Lecture Notes in Networks and Systems, Proceedings of the Innovations in Computer Science and Engineering, Singapore, 16–17 August 2019*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 295–305.

14. Agwa, N.A.; Kobayashi, T.; Sugimoto, C.; Kohno, R. Security of Patient's Privacy in E-Health using Secret Sharing and Homomorphism Encryption Scheme. In Proceedings of the 2020 35th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Nagoya, Japan, 3–6 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 155–160.
15. Sun, X.; Yu, F.R.; Zhang, P.; Xie, W.; Peng, X. A survey on secure computation based on homomorphic Encryption in vehicular Ad Hoc networks. *Sensors* **2020**, *20*, 4253. [CrossRef] [PubMed]
16. Zhou, S.; Yu, Z.; Nasr, E.S.A.; Mahmoud, H.A.; Awwad, E.M.; Wu, N. Homomorphic encryption of supervisory control systems using automata. *IEEE Access* **2020**, *8*, 147185–147198. [CrossRef]
17. Cominetti, E.L.; Simplicio, M.A. Fast additive partially homomorphic Encryption from the approximate common divisor problem. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 2988–2998. [CrossRef]
18. Acar, A.; Aksu, H.; Uluagac, A.S.; Conti, M. A survey on homomorphic encryption schemes: Theory and implementation. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 1–35. [CrossRef]
19. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
20. Chen, Y.C.; Hung, T.H.; Hsieh, S.H.; Shiu, C.W. A new reversible data hiding in encrypted image based on multi-secret sharing and lightweight cryptographic algorithms. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 3332–3343. [CrossRef]
21. Lin, H.; Garg, S.; Hu, J.; Wang, X.; Piran, M.J.; Hossain, M.S. Privacy-enhanced data fusion for COVID-19 applications in intelligent Internet of medical Things. *IEEE Internet Things J.* **2020**, *1*. [CrossRef]
22. Vizitiu, A.; Niţă, C.I.; Puiu, A.; Suciu, C.; Itu, L.M. Applying deep neural networks over homomorphic encrypted medical data. *Comput. Math. Methods Med.* **2020**, *2020*, 26. [CrossRef] [PubMed]
23. Cheng, W.; Ou, W.; Yin, X.; Yan, W.; Liu, D.; Liu, C. A Privacy-Protection Model for Patients. *Secur. Commun. Netw.* **2020**, *2020*, 12. [CrossRef]
24. Yang, Y.; Xiao, X.; Cai, X.; Zhang, W. A secure and high visual-quality framework for medical images by contrast-enhancement reversible data hiding and homomorphic Encryption. *IEEE Access* **2019**, *7*, 96900–96911. [CrossRef]
25. Cano, M.D.; Cañavate-Sanchez, A. Preserving data privacy in the internet of medical things using dual signature ECDSA. *Secur. Commun. Netw.* **2020**, *2020*, 4960964. [CrossRef]
26. Nguyen, D.C.; Pathirana, P.N.; Ding, M.; Seneviratne, A. BEdgeHealth: A Decentralized Architecture for Edge-based IoMT Networks Using Blockchain. *IEEE Internet Things J.* **2021**, *8*, 11743–11757. [CrossRef]
27. Qiu, T.; Liu, X.; Han, M.; Ning, H.; Wu, D.O. A secure time synchronization protocol against fake timestamps for large-scale internet of things. *IEEE Internet Things J.* **2017**, *4*, 1879–1889. [CrossRef]