sustainability

MDPI

Article

A Novel Method to Enhance Sustainable Systems Security in Cloud Computing Based on the Combination of Encryption and Data Mining

Qian He [1,*] and Hong He [2]

[1] Enterprise Cooperation and International Affairs Office, Hunan Industry Polytechnic, Changsha 410007, China
[2] Department of Computer and Communication, Hunan Institute of Engineering, Xiangtan 411101, China; hehong@hnie.edu.cn
* Correspondence: heqianheqianhe@gmail.com

Abstract: Due to the increasing growth of technologies and the diversity of user needs in the field of information technology, the position of cloud computing is becoming more apparent. The development of computing infrastructure in any organization requires spending a lot of money, time, and manpower, which sometimes does not fit into the operational capacity of an organization. Therefore, organizations tend to use such technologies to advance their goals. A fully open and distributed structure in cloud computing and its services makes it an attractive target for attackers. This structure includes multiple service-oriented and distributed paradigms, multiple leases, multiple domains, and multi-user autonomous management structures that are more prone to security threats and vulnerabilities. In this paper, the basic concepts of the cloud computing and its applications have been investigated regarding to the importance of security issues. The proposed algorithm improves the level of security in the cloud computing platform through data mining and decision tree algorithm. Low computational burden and client numbers independency help to effectively implement the proposed algorithm in reality.

Keywords: cloud computing; security; data mining; decision tree

Citation: He, Q.; He, H. A Novel Method to Enhance Sustainable Systems Security in Cloud Computing Based on the Combination of Encryption and Data Mining. *Sustainability* **2021**, *13*, 101. https://dx.doi.org/10.3390/su13010101

Received: 9 November 2020
Accepted: 15 December 2020
Published: 24 December 2020

1. Introduction

Today, the field of cloud computing has received a lot of attention due to its many benefits. These benefits include good costs, time savings, and optimal use of computing resources. But the issue of security and privacy are some of the issues that have made this phenomenon viewed with skepticism. The nature of cloud computing requires users to transfer their data to cloud service servers. Due to the mechanism of security issues of a cloud system on the server side, data owners may not be aware of the challenges and solutions adopted on the server side. Some studies have focused on topics such as server-side application security, operating system security, virtual machine security, or hardware-level security issues. These studies have usually not been comprehensive and also the issues related to users' data security are only under the control of the server itself. Another approach to securing users' data is to use technologies provided by third-party companies called trust computing. Although these approaches allow users to monitor and evaluate the security aspects of their data using specific tools, they do not provide the user with much control over the practical aspects of security issues. Data-centric security is a new approach to giving users the ability to control data security from within during the time the data are on the server. However, due to the novelty of this approach, a standard framework for data-centric security methods in cloud-based systems has not yet been provided. In this study, we intended to improve user data security on cloud servers by providing a data-centric security-based solution that provides users with an appropriate level of control and assurance of data security.

In this paper, the presented innovation and contributions can be expressed as follows:

- Combining decision tree data mining method with cryptographic security protocols.
- Providing an asymmetric pattern for coding considering the limited loading conditions.

2. Literature Review

The world of information technology and the Internet, which has become a vital part of human life today, is expanding day by day. At the same time, the needs of community members, such as information security, fast processing, dynamic and instant access, the ability to focus on enterprise projects instead of wasting time on server maintenance, and, most importantly, cost savings, have become increasingly important. The solution that is proposed in the field of technology for such problems today is a technology that these days is called cloud computing. Cloud computing is a model for having comprehensive, easy, and customizable network access to a set of configurable processing resources (e.g., networks, servers, storage space, applications, and services) that can be provided quickly or released with the least work and management effort or the need for the intervention of the service provider [1].

In short, through corporate cloud computing, IT service users can purchase their IT-related services as a service. This means that instead of buying servers for internal or external services, or purchasing software licenses, companies can buy them as services. Cloud computing is a way to increase storage capacity or facilities without spending money on new infrastructure, training new staff, or purchasing new software licenses. In fact, companies or individuals will only pay for what they consume [2]. So it is an effective way to use resources, capital management, and technology support costs. Several works have been done to optimize the data center's performance [3–6]. A temporal task scheduling method has been developed in [7] to study the variation of temporal in hybrid cloud with consideration of their delay constraints. In [8] a logarithmic operation has been proposed to reduce the standard deviation before smoothing workload and resource sequences in cloud systems.

Security of cloud computing (sometimes referred to as cloud security) is a minor field of computer security, network security, and, more broadly, information security [9,10]. This concept includes a set of technologies, policies, and controls to protect applications, data, and security infrastructure in cloud computing.

Cloud computing is a very meaningful concept and the idea is very simple. Instead of storing your data on a PC, you will store it on a server on the Internet. It does not matter to you where this computer is located. It is possible for your data to be stored on a large number of computers and not just one computer. As long as you are connected to the Internet and have enough bandwidth, you can transfer your photos, files, and even videos to the desired server using the device you want, such as a mobile phone, a quality computer, or Internet kiosks at the airport [11].

Cloud computing is still a hot topic. However, even with all the discussion of cloud computing, there still seems to be a lot of debate in defining a set of cloud computing. There are many definitions for cloud computing [1]:

- In general, it is a style of computing where IT-related capabilities are provided as a service using Internet technologies to external customers.
- The emergence of IT presentation and development models that lead to the immediate delivery of products, services, and solutions through the Internet.
- A service model that provides the principles of general production for the delivery of information technology, infrastructure components, architectural methods, and a model based on economic principles, which is related to the concepts of computational grade, virtualization, utility computing, hosting, and presentation of software as a service.

- The idea of providing personal applications (such as email, text presentation, and processing) and business benefits (such as accounting, automation of the sales system, and customer service) of centralized servers.

The emergence of basic concepts of cloud computing dates back to the 1960s and when John McCarthy stated that "Computing may someday be organized as a public utility just as the telephone system is a public utility". The term cloud is actually derived from the telephone industry, whereby telecommunications companies, which until the 1660s offered only dedicated point-to-point lines, began to offer virtual private networks of similar quality and lower cost. The cloud symbol represents the boundary point between the parts that are in the scope of the user's responsibility and those that are used in the scope of the supplier's responsibility. Cloud computing extends the concept of cloud to include servers in addition to network infrastructure. Amazon has played an important role in the development of cloud computing by modernizing its data center [1].

Relative security of cloud computing is a controversial issue that may delay the full acceptance of cloud computing in public opinion and lead to reduction in the required efficiency. One group believes that data security is higher when managed within the organization, while another group believes that service providers have a strong motivation to maintain trust and therefore use a higher level of security. In addition to data security, the availability and performance of cloud-hosted applications is also important to users. Privacy advocates also criticize the cloud model, as cloud service providers can have full legal or illegal control over data in communications between service users and the cloud host.

In the present century, we have been witnessing an increase in the use of lightweight portable devices to access Internet services instead of personal computers. Since such devices do not have strong processing capabilities, then who will provide the necessary processing power? The answer to this question lies in cloud computing.

The cloud computing is broadly going to use in different sustainable systems. One of the main applications on cloud computing is the implementation of smart grid protection and control in the cloud. Moreover, the security of cloud computing is a critical issue in cyber security concerns of the grid [12].

3. Background and Challenges

Studies have shown that among the major challenges in cloud computing, security is the biggest and most important challenge. Since cloud computing encompasses many technologies, including networks, databases, operating systems, resource scheduling, transaction management, concurrency control, and memory management, different security threats can be imagined according to different security requirements (confidentiality, integrity, access control, privacy, etc.).

In this section, we examine the challenges of data security and privacy by emphasizing the principle of information confidentiality in the process of processing users' data in the cloud. In the next section, we will classify and evaluate the research conducted in this field. Some of the most challenging research issues in cloud computing is as follows:

- Emergence of various security needs for organizations by continuing to outsource information security affairs to the outside world.
- The process of outsourcing data in cloud servers by service providers and secure access to data and information outsourced during computing, because data outsourcing in the cloud prevents easy and managed user access to personal information.
- The process of encrypting and decrypting data on cloud servers can be challenging when performing computing on large volumes of data stored in the cloud. It is also a type of cryptographic process that protects the data during the transfer from risks and can ensure the security of the computations during the processing stage.
- Ensuring the implementation of appropriate policies in data sharing in the cloud network

- SLA (service level agreements): vendors of cloud computing services resist legal action and instructions, such as the cost of computing services or outsourcing, data protection, etc., and violate the rules imposed on data privacy, while reassuring their customers to protect their privacy and data and advising them to follow the rules. This can also be challenging.
- Managing a huge volume of data in the cloud is very difficult and requires the establishment of systems based on intelligent management of distributed databases. As in this issue, the most important challenge is timely availability with high efficiency, maintaining confidentiality, and secure data transfer during outsourcing.
- Access controls: Authentication management is an important issue and is based on the security levels of passwords and the frequency of their periodic changes, by which the user must summon the service provider. Additionally, determining methods for recovering passwords and the ability to audit different accesses to secret data are all challenges that need to be addressed.

Despite the above, the following two key questions, as the most recent challenges in the field of cloud computing security, have received special attention from researchers:

- How can we better ensure data security? (Refers to how to fix the shortcomings of the issue of information leakage and breach of confidentiality).
- How can the private information of a cloud-based client be kept completely confidential?

However, in a study [2] published by the US National Security Agency in collaboration with two other organizations entitled "Securing the Cloud with Homomorphic Encryption", two other important issues have been raised as a new challenge in this area:

- Search in encrypted data;
- Sharing keys.

These two cases can be explained with two examples:

Suppose a person in New York City who uses cloud infrastructure to send and receive emails wants to search their emails for an order and send an order to a New Jersey car dealer.

The challenge here is that if the data are sent openly, the subscriber enters the search term and sees the result, and if the data are encrypted, the person will need to share their private secret key to access the information with the service provider.

In this way, key sharing creates the intrusion potential. In accessing shared data, an intruder can access the data if they want to.

In the next section, in the ideas and solutions section, we will see that "complete homomorphic encryption" enables the use of the Hoboken Mishap technique for the hypothetical person in the example above to search for the phrase they want among the encrypted emails without sharing the key in the insecure cloud, producing the same result as if they shared the key or sent the text publicly.

4. Architecture of a Cloud Computing System

In general, the cloud processing architecture consists of three layers [13]:

- Software as a service—SaaS;
- Platform as a service—PaaS;
- Infrastructure as a service.

The general layers of a cloud computing service are shown in Figure 1, but actually and in more detail, this architecture contains five layers:

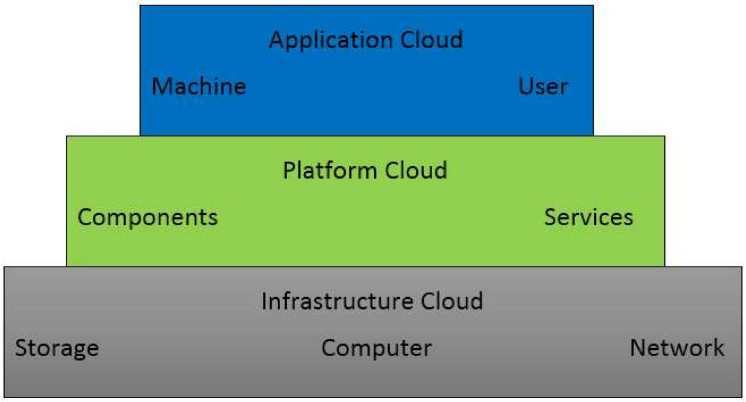

Figure 1. Layers of a cloud computing service.

4.1. First Layer: User

Cloud computing including computer software and/or computer hardware that relies on cloud computing for application delivery, or that is specifically designed for delivery of cloud services and that, in either case, is essentially useless without it. In general, the characteristics of this layer are:

- Software or hardware;
- Receiver of cloud services;
- Inefficiency without cloud services.

These include phones with iOS, Android, and Windows Mobile operating systems; lightweight users such as Zonbu, CherryPal, and GOS (Global Observing System)-based systems; large users and web browsers such as Firefox, Google Chrome and WebKit [14].

4.2. Second Layer: Software as a Service

Cloud application services, or software as a service (SaaS), delivers software as a service on the Internet, thereby eliminating the need to install software on customer computers and making the maintenance and support easier. The main features of these services are:

- Access and management of commercial software through the network [15]. Activities are managed by specific centers, and these centers are somewhere other than the location of each customer and, as a result, customers can access the applications remotely and via the web. The software delivery model is closer to a one-to-many model (a running version of the application a multitenancy model) than the one-to-one model [11].
- Software updates and upgrades are managed centrally and eliminate the need to download patches or upgrades. P2P software such as Skype; web applications such as Facebook, Twitter, and YouTube; security services such as MessageLabs; software services such as IBM Lotus, Google Labs, Gmail, salesforce.com, Payroll, Google Calendar, CRM, HR, and Live; software add-on services such as Microsoft Online Services; and storage services are examples of application layers.

The above companies have been established specifically for the software trade as services, which receive a fee for the registration of their users and their software is installed on their central servers, while users access the application via the Internet [16].

4.3. Third Layer: Platform as a Service

It provides cloud platform services or platform as a service, computing platform, or solution platform (often running on cloud infrastructure that feeds the cloud application) as a service. In this model, instead of platform software, it is like a service [17].

With platform as a service, software can be expanded without spending money and complexity in purchasing and managing major hardware and software, as well as providing

web hosting facilities, and software developers do not need to spend development costs to create new applications or develop old ones [1].

This service provides a layer of software that can be used to produce higher level services. Platform service includes, middleware, integration facilities, exchange of message, information, and connection setup.

A good example would be the Google Apps software development engine, which allows applications to run under Google's infrastructure. Platform services like these can provide us with powerful fundamental facilities for developing applications, and of course the facilities provided to the developer can be limited by the service provider. For example, an application generated by the Google Apps engine is ultimately owned by Google, and we cannot provide the end user with features beyond what Google has put in this engine (package). Various companies have developed platforms that allow the end user to run applications through centralized servers via the Internet. Azure operating system from Microsoft and Google Apps are examples of these services.

4.4. Fourth Layer: Infrastructure as a Service

Cloud infrastructure services, or infrastructure as a service, provide computer infrastructure (usually a virtual platform) as a service. Instead of buying hardware and software and data center space or network equipment, users buy all of this infrastructure as a fully outsourced service. Service bills are usually issued based on the public computing model and the amount of resources consumed, and therefore the cost reflects the amount of activity. This method is in fact an evolution of the virtual private server supply model, which is often a virtual computing environment.

In general, with infrastructure as a service, the computer infrastructure and the platform virtualization environment can be provided as a service. Amazon AWS service is one such example [18].

4.5. Fifth Layer: Server

It includes the physical part of the cloud and refers to hardware or software that is specifically dedicated to cloud computing services. Examples include multicore processors and cloud-specific operating systems.

These five layers make up the overall architecture of cloud computing. In the meantime, users may investigate more in the presence and progress of the first and second layers, while the rest of the layers are also very important for service providers and developers [19–23].

5. Security in Cloud Computing

In today's computer systems, anyone who wants to analyze the information contained in encrypted data must first decrypt the data, in which case there is a risk of cyberattack by hackers and misuse of information. When data are transmitted over the Internet, maintaining its security in the cloud is a major concern. Thus, information stored in the cloud also requires protection through standard encryption techniques [24–28].

On the other hand, in cryptographic systems, the second party or the receiver must have the sender's private key to decrypt the information. Therefore, every time a client sends a request to its "virtual environment" in the cloud and expects fast and secure computing on its data, a private key must be provided (by sending along with the user's request to the cloud) and the computing is done after data decoding. However, this will increase the risk of the key being leaked each time it is processed, in which case the client will have to change the key, and if symmetric encryption is used, both parties must have the same secret key and in case of leakage, they must generate the secret key again. Despite the high speed, while violating security, this will increase processing time and computational overhead.

Confidentiality can solve the problem. The comprehensive solution and technique that maintains data confidentiality is "cryptography". But what kind of cryptography can safely analyze encrypted information? Which is suitable, symmetric or asymmetric?

Data security includes encryption and ensuring that appropriate cloud sharing policies are implemented. Cloud computing is a new computing model that originates from network computing, distributed computing, parallel computing, virtualization technology, applicable computing, and other computer technologies. This system has several special advantages, which include: big data storage and computing, virtualization of most hardware services, high scalability and reliability, and low cost. This type of computing is basically the idea that all data and applications (which are stored and installed centrally in data centers and cloud servers) can be used from anywhere in the world via the Internet and laptops and mobile phones or any online device.

Every organization needs its own system for identity management to control the access to computing resources and information. Cloud computing providers or customer identity management systems integrate their infrastructure with SSO (Single Sign-On) technology or provide a proprietary identity management solution [4]. For the sake of personal and physical security, providers guarantee that physical machines are secure enough that access to these machines and customer data are not the only limitation, and that all access is documented. For availability, providers ensure regular and predictable access to their applications and data.

Providers guarantee that the applications are available as a service on cloud computing, whose security is achieved by implementing testing and acceptance procedures for sending out or coding packaged applications. This mechanism also requires criteria for the security of applications in the client environment. Finally, for privatization purposes, cloud computing providers ensure that all sensitive data (for example, credit card numbers) are hidden and only authorized users are allowed to access the data. In addition, digital IDs and certificates must be protected from any data that the provider collects or generates about the customer's activities. Also, customers and providers should consider the legal issues, such as E-Discovery and contracts and related laws, that may change from country to country [13].

The cloud security architecture just works when the right defense implementations are in place. An efficient architecture of cloud security must identify security issues at the management level. Security management addresses security control issues. These controls are designed to protect against any vulnerabilities in the system and reduce the impact of an attack.

Although there are many types of controls behind the cloud computing security architecture, they can fall into the following categories [14]:

- Deterrent controls

Deterrent controls are configured to prevent any intentional attack in a cloud computing system. These controls do not reduce the actual vulnerability of a system.

- Preventive controls

These controls increase the strength of the system by managing vulnerabilities. Preventive control will protect the system vulnerabilities if an attack occurs, then this type of control will try to cover the attack and reduce system security failure.

- Corrective controls

This control tries to reduce the effect of the attack. Unlike preventive control, corrective control reacts when an attack occurs.

- Detective controls

This type of control tries to detect when an attack occurs. At the time of the attack, the detection control sends a signal to preventive or corrective controls to identify the problem.

6. Data Mining

New information and communication technologies, as well as decision support technologies, by collecting, storing, evaluating, interpreting, retrieving, and disseminating

information and knowledge to specific users, can have a great impact on finding timely, accurate, and needed information for people. One of the tools that are applied in these technologies is data mining. Data mining consists of implementing tools of advanced data analysis to figure out valid, previously unknown patterns and relationships in a large set of data. These tools are mathematical algorithms, statistical models, and machine learning methods (methods that automatically enhance their performance based on learned experiences). Data mining goes beyond data collection and management, and includes analysis and forecasting. Another name for it is knowledge discovery in database or KDD for short.

Data mining can be done on textual, quantitative, or multimedia data. Its applications include the following:

- Association rule: patterns in which the existence of one item implies the existence of another item;
- Classification: attributing patterns to a small set of predefined classes by discovering some relationships between properties;
- Clustering: grouping of customers or sets of patterns that have similar features;
- Prediction: discovering patterns to make logical predictions about the future;
- Path analysis or sequential patterns: patterns in which one event leads to another.

Data mining is not a new technology, but its application is growing significantly in various private and public sectors, and generally industries such as banking, insurance, medicine, and retail use data mining to reduce costs, increase research, and increase sales.

Data Mining Operations

In data mining, four main operations are performed, which are:

- Predictive modeling;
- Database segmentation;
- Link analysis;
- Deviation detection.

Of these main operations, one or more of them are used in the implementation of various data mining applications. For retail applications, for example, segmentation and link analysis operations are commonly used, while any of the four operations can be used to detect fraud. In addition, a sequence of operations can be used for a specific purpose. For example, to identify customers, the database is first segmented and then predictive modeling is applied to the created parts [29].

Data mining techniques, methods, and algorithms are ways of implementing data mining operations. Although each operation has its strengths and weaknesses, various data mining tools select operations based on specific criteria. These criteria are:

- Proportion to the type of input data;
- Transparency of data mining output;
- Resistance to errors in data values;
- Output accuracy;
- Ability to work with large volumes of data.

7. The Proposed Model

In this paper and in the proposed modeling, we will improve the security of cloud computing by presenting an innovation approach based on a combination of the decision tree method and cryptography through cryptographic protocols. This model is designed to first identify and understand the limitations of the decision tree. Then, after defining the users in the decision tree, it will be time to describe the public key in the system. Asymmetric cryptography, based on identifying the limitations that previously existed in the decision tree, provides the public key to the authorized user and after identifying the authorized user, the private key will also be provided to the user. In the following, we will present the algorithms used.

7.1. Decision Tree

Selection trees are an approach for representing a set of rules that lead to a value or class. For instance, someone might want to categorize loan applications as good or bad credit risk. Figure 2 shows a simple model of a selection tree with an explanation of all its base packages, i.e., the selection node, its branches, and leaves, which solves this problem [30].

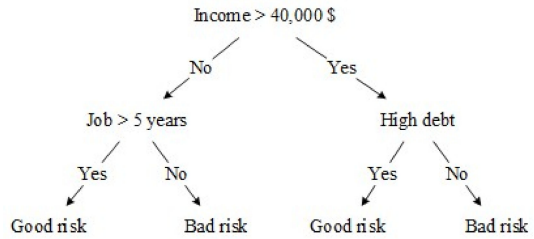

Figure 2. A simple model of a decision tree.

The first package is the top node of the decision or root, which is a check to establish a specific situation. The root node is "Income > $40,000" in this example. The outcome of this study leads to the tree being separated into two branches, and each of which determines the possible answers. In this case, the study of the mentioned situation can have a yes or no answer, so we have two branches.

Depending on the algorithm type, each node can have two or more branches. For instance, CART makes trees with just two branches per node. Such a tree is introduced as a binary tree.

Several decision tree models are commonly presented in data mining to explore data and to deduce the tree and its rules used for the prediction. A number of approaches can be used to build decision trees, including CHAID, CART, Quest, and C5.0.

Tree size can be controlled by stopping laws that restrict tree growth.

7.2. Decision Tree Learning

The structure of the decision tree in machine learning is a predictive model that applies the observed facts about a phenomenon to inferences about the target value of that phenomenon. The machine learning technique for inferring a decision tree from data is called decision tree learning, which is one of the most common methods of data mining.

Each internal node corresponds to a variable and each arc to a child represents a possible value for that variable. A leaf node, with values of variables represented by the path from the root of the tree to that leaf node, represents the predicted value of the target variable. A tree decision represents a structure in which the leaves represent the classification and branches represent the seasonal composition of the attributes that result from these categories. Learning a tree can be done by breaking a resource set into subsets based on an attribute value test. This process is repeated recursively in each subset of the segregation. The return operation is complete when further segregation is no longer useful or a classification can be applied to all samples in the resulting subset.

Decision trees are able to generate human-understandable descriptions of relationships in a data set and can be used for classification and prediction tasks. This technique has been widely used in various fields, such as plant disease diagnosis and customer marketing strategies.

This decision structure can also be introduced in the form of mathematical and computational techniques that help describe, categorize, and generalize a set of data. Data are given in records in the form $(x, y) = (x_1, x_2, x_3 \ldots, x_k, y)$. Using the variables $x_1, x_2, ..., x_k$, we try to understand, categorize, or generalize the dependent variable Y.

7.2.1. Functions Used in the Decision Tree

❖ Classregtree

This function creates a correction function to predict the y response as a predictor function in the X columns. X is an m×n matrix of the predicted values. If y is a vector of the response values of n, the classregtree performs the regression. If y is a batch variable, character array, or cell array of character vectors, classregtree performs the classification. In either case, t is a binary tree in which each radial node is divided by the values in column X. NaN values in X or Y are taken as missing values. The observations do not fit all missing values for X or missing values for y. Observations with some missing values for X are used to find division on variables that these observations have valid values.

❖ Decision tree performance

The decision tree assigns a recognizable value and attribute to each of them according to the information and characters imported. According to these characteristics, the decision tree starts the construction of the equivalent graph and forms a graph dedicated to the data based on the attributes and using the internal yes/no algorithm.

7.2.2. Introducing the Idea of Homomorphic Encryption and Providing Solutions

The Gentry system supports all addition and multiplication operations in encrypted data operations, but was it possible to build circuits to perform arbitrary computation in addition to addition and multiplication? Thus, a somewhat homomorphic encryption (SHE) scheme was proposed that could process up to a maximum of low-order polynomials on an encrypted message. This system was limited because it propagated noise. As we will see in future sections, noise could increase with the expansion of processing operations to the extent that decoding became impossible.

As we have seen, Gentry changed the system a bit and used the idea of bootstrap, and finally proved that any bootstrappable somewhat homomorphic encryption approach can be transformed to a fully homomorphic encryption using a recursive self-embedding. It should be noticed that, the security of the Gentry system had two major problems:

- It was difficult to implement.
- In some cases, the worst case was based on incapable lattice.

In 2010, four researchers, Craig Gentry, Shai Halevi, Marten van Dijk, and Vinod Vaikuntanathan, worked on a fully homomorphic encryption scheme. They explored Gentry's ideas and found that the system did not require lattice methods. The researchers showed that somewhat homomorphic component of Gentry's ideal lattice-based scheme could be exchanged by a simple somewhat homomorphic approach that utilized integers, which eventually led to the emergence of the second generation of homomorphy. Some other scientists, such as Levieil and Naccache, came up with schemes and improvements to the algorithm based on Mr. Cohen's ideas. Some of this research has become experimental implementation and some practical.

7.3. Second Generation of Full Homomorphy

New methods were developed from 2011–2012 by three researchers named Vika Brakerski, Craig Gentry, and Vinod Vaikuntanathan, including:

- BGV (Brakerski-Gentry-Vaikuntanathan) encryption system;
- Brakerski unlimited scale cryptography system;
- NTRU-based encryption systems.

GSW (Gentry-Sahai-Waters) encryption system: the security of some of these systems (second generation of full homomorphy) is based on learning difficulty with error problem, with the exception of the LTV (Lopez-Alt, Tromer, and Vaikuntanathan) system, where security is based on a variable of the NTRU (Nth degree-truncated polynomial ring units) computational problem. A distinctive feature of this encryption system is that with all their features, noise grows much less during homomorphic computation.

The features of the GSW encryption system make this system grow with less noise, so it has stronger security and better efficiency. All of the second-generation methods followed the original Gantry scheme and eliminated the noise effect by improving the SHE system, and then converted it to the bootstrapping FHE system

7.4. Improving Keys and Security of the Files

In order to improve security of files in the proposed method, the owner of a data should utilize a specified symmetric key *Ks* to encrypt the file before the sending process to store it in the cloud. Referring to the DCS (Data Centric Security) method, all of the security requirements are connected to the data, so each symmetric key is securely attached to the file.

$$X \equiv a_1 mod\ n_1 X \equiv a_2 mod\ n_2. \tag{1}$$

The encrypted text generated by $C_r||K_s$ for each user is used to replace the value of a_i in Equation (1) to generate the following congruence:

$$X_r \equiv E_{Kpub_1}(Cr\Big|\Big|Ks\Big)\ mod\ n_1\ X_r \equiv E_{Kpub_2}(Cr\Big|\Big|Ks\Big)\ mod\ n_2 \vdots X_r \equiv E_{\mathrm{Kpub}_k}(\mathrm{Cr}\Big|\Big|\mathrm{Ks})\ mod\ n_{\mathrm{k.}} \tag{2}$$

The solution of this congruence X_r, in which $0 \leq Xr < n = n_1 n_2 \ldots n_k$, is the shared value of the resource *r* and is attached to that resource, and both of the shared value and that resource are kept on a cloud server together. When a user with authority u_i as user demands to have access to the resource *r*, the cloud server transmits the *Xr* (shared value) to the user. When the user receives the X_r (shared value), the user utilizes the corresponding private key to decrypt *Xr* to generate $Cr||Ks$ shown in the following equation

$$C_r\Big|\Big|K_s = D_{K_{prv\ i}}(X_r mod\ n_i) \tag{3}$$

In the above equation, the value of $D_{Kprv\ i}$ is the decryption operation based on the user's private key and n_i is the first number assigned to the relevant user. The C_r value is returned by the user to the server to check the user's permission and give them access to the desired resource. In the next step, the server sends the desired source to the user and the user utilizes the K_s to decrypt and access content of the desired file.

In the equation, two symmetric keys K_s are protected in two levels. In the first level they are encrypted by an authorized user and then in the second level they are encoded by the decision tree to find the common value of X_r. The *Ks* can be obtained using Equation (3) from X_r and only by the authorized users, which requires the n_i and an authorized user private key. The data owner adds themselves as a user for optimal key management when calculating X_r by using Equation (2) for each file. Therefore, the data owner and other users need to store the K_s value and only need to store the private key and the assigned prime number (n_i). Consequently, the Ks key is shared in a secure and optimal way between authorized users and also is preserved from the access of the unauthorized user, including the file server itself.

The value of *Cr* and also the secret key of K_s are both specified in all files. If the value for one the files is compromised, all of the other remaining files retain secure. The value of C_r, which preserves the secrecy, is utilized by a user with authority to prove permission and access the relevant resource. The data owner appends securely the secret value of C_r to the corresponding file and in addition transmits it to the server inside the X_r encrypted text. The secret value of C_r can be calculated using the shared value of X_r only by the users with authority. Therefore, the C_r value can be detected only by the authorized users for the server and prove that they are actually allowed to access the source. The secret value of C_r is unique to each file, even the same user uses a different one for different files. That way, if a C_r is compromised for a particular file, the other files will not be affected. In addition, this feature can be used when the data owner wants to change the details of the authorized users for that file. For example, to add a new user with authority to a particular file, the owner only need to update the X_r value for the file. Since the C_r parameter can remain constant, there is no need to resend it to the server. This reduces the risk of C_r

being compromised, while keeping a dynamic mechanism to update the list of the users with authority. Although, if the access of a user should be denied to certain data that were previously allowed to be accessed, the C_r value must be changed to maintain security issues.

8. Simulation Results

Figure 3 shows a comparison between the method presented in this paper and other debatable methods that show the time overhead for processing files on the server side. As can be seen, the method presented in this paper is somewhat weaker than the methods mentioned in this paper. This makes perfect sense given the time overhead of creating a DCS file:

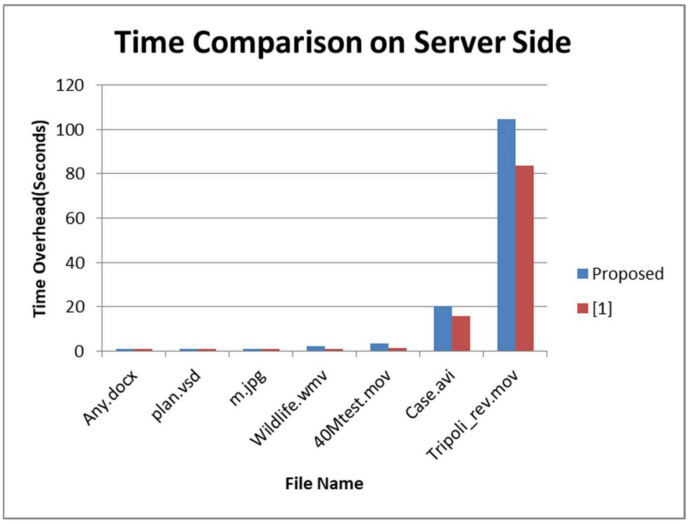

Figure 3. Comparison of time overhead of two methods on the server side.

However, on the other hand, the results illustrated in Figure 4 show that access to data on the client side is much faster because the mathematical calculations performed on the client side are much more complex than the other methods presented. In addition to these issues, the user's complete control over data encryption policies should be considered, which was not considered in the older method, and the main burden of control is on the server, which may not involve user's satisfaction of different access levels. It should be noticed that the presented times are the average time overhead for different cases.

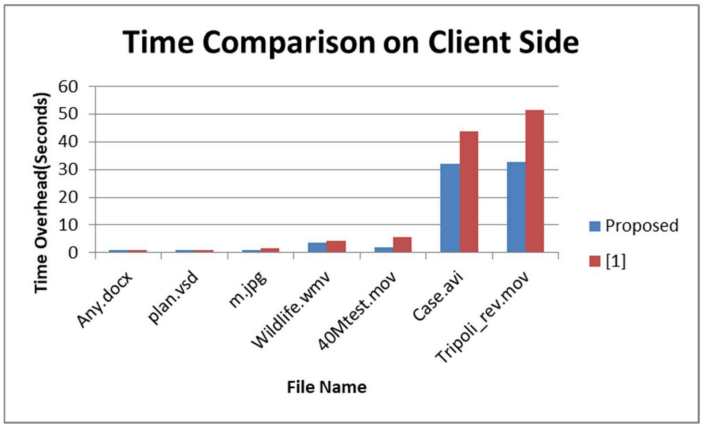

Figure 4. Comparison of time overhead of two methods on the client side.

On the other hand, the decoding time criterion is also very important. A particular method may greatly reduce the time and memory overheads, but the time overhead

imposed by decryption (which is an unavoidable part of the system) makes it impossible to use. Table 1 shows the size of the files in different cases. It can be seen that the maximum file size is for the .mov files, which contain large file sizes. Therefore, the proposed algorithm can be applicable for even large sizes of files. This superiority helps to implement the proposed algorithm in reality. Also, Figures 5 and 6 show the time overhead of the proposed algorithm when each case is considered simultaneously. This means that all the files in each case become accessed by the client in the same time.

Table 1. The file size (mb) of the comparison results for different cases.

Cases	docx	vsd	jpg	wmv	mov	avi	mov
1	1.1	4.7	6.7	15	40.7	190	700.9
2	2	5	7.9	18.2	46.5	210.5	851.6
3	3.5	3.2	9.8	20.7	59	499	938.7
4	0.5	6.7	10	26.8	71.9	370.8	866.8

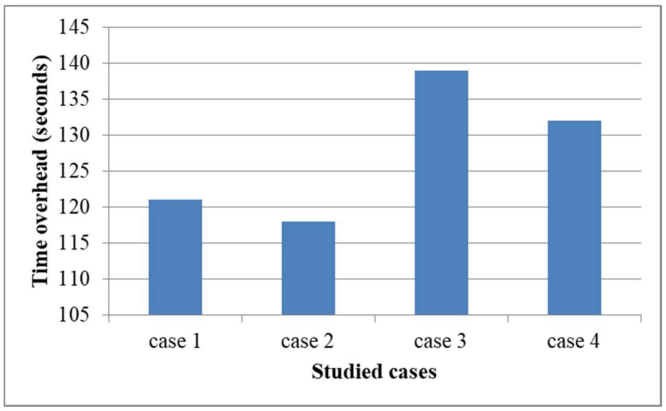

Figure 5. Time overhead of all cases on the server side.

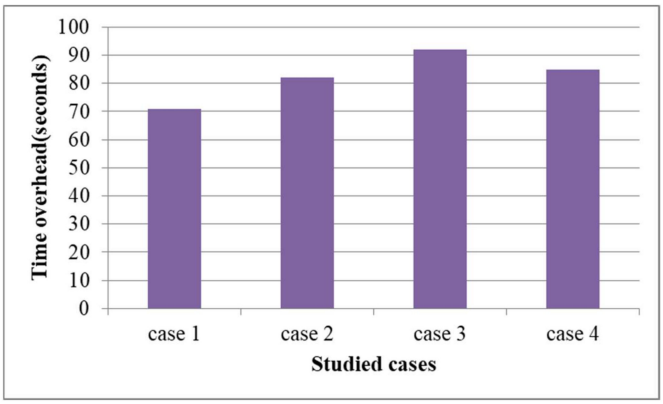

Figure 6. Time overhead of all cases on the client side.

It can be seen that the most changes in time overhead occur on the server side and the changes in time overhead on the client size are small. It means that the client will lose much less time and this time is somehow in an acceptable range. However, the time overhead on the server side can be increased or decreased significantly based on the cases.

To provide more comparison for the performance of the proposed method with the recent works, Figure 7 shows the total runtimes for the proposed method and a recent published work with different client numbers. It can be seen that the total run time of the proposed method is changed slightly by increasing the number of clients, which shows the simplicity of the proposed algorithm and computational burden. Whereas, the total

runtime for other relative works can be increased significantly when the numbers of clients become increased.

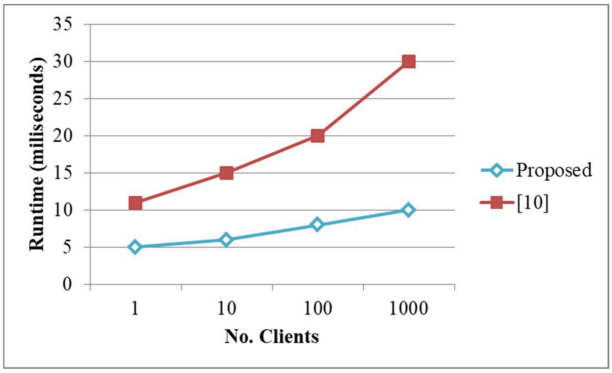

Figure 7. Comparison of runtime of two methods with different numbers of clients.

9. Conclusions

An important point about the proposed algorithm and the evaluations made is its weakness in terms of dependency and the need to build a new data structure and computation, time, and memory overheads imposed on the server. This is because, by default, we intended to consider the server as an unauthorized entity in accessing the data. Thus, we needed each file to have all the additional data needed and no need for server to decrypt it. This makes us accept the overhead caused by this data structure in all cases. On the other hand, the main purpose of this paper was to give the user complete control over data protection, but in the defined system, the server is always considered as an unauthorized entity, so contrary to popular belief, the level of access in which the server can read the data is not defined. Therefore, this paper intended to introduce a hybrid method to reduce computational and memory overheads if there is no need to hide the data from the server. In this method, we also consider the case that instead of defining the authorized users to access the data, unauthorized users are assigned access. This is because sometimes the data owner may want to hide the data from only a handful of people and allow others to access the data. Therefore, in such a case, defining the list of unauthorized people is more logical than the list of authorized people, whose number is very large.

On the other hand, it should be noted that in many applications, the data owner may not know all the authorized users, but the user in question is actually authorized to access the data. For example, if we assume that all users in a geographic area can access the data, the data owner certainly does not know all of these users, or the intended user may not exist at all at the time the file is generated (not connected to the network).

Author Contributions: Conceptualization, Q.H. and H.H.; methodology, Q.H. and H.H.; software, Q.H. and H.H.; validation, Q.H. and H.H.; formal analysis, Q.H. and H.H.; investigation, Q.H. and H.H.; resources, Q.H. and H.H.; data curation, Q.H. and H.H.; writing—original draft preparation, Q.H. and H.H.; writing—review and editing, Q.H. and H.H.; visualization, Q.H. and H.H.; supervision, Q.H. and H.H.; project administration, Q.H. and H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Union Fund in Hunan Province and Xiangtan City of China grant number (2019JJ60041) and the APC was funded by the Union Fund in Hunan Province and Xiangtan City.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationship that could have appeared to influence the work reported in this paper.

References

1. Vaquero, L.; Rodero-Merino, L.; Caceres, J.; Lindner, M. A break in the clouds: Towards a cloud definition. *ACM SIGCOMM Comput. Commun. Rev.* **2008**. [CrossRef]
2. Che, J.; Duan, Y.; Zhang, T.; Fan, J. Study on the security models and strategies of cloud computing. *Procedia Eng.* **2011**, *23*, 586–593. [CrossRef]
3. Yuan, H.; Bi, J.; Li, B.H.; Tan, W. Cost-aware request routing in multi-geography cloud data centres using software-defined networking. *Enterp. Inf. Syst.* **2017**, *11*, 359–388. [CrossRef]
4. Yuan, H.; Bi, J.; Zhou, M.C. Spatiotemporal Task Scheduling for Heterogeneous Delay-Tolerant Applications in Distributed Green Data Centers. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 1686–1697. [CrossRef]
5. Yuan, H.; Bi, J.; Zhou, M.C. Profit-Sensitive Spatial Scheduling of Multi-Application Tasks in Distributed Green Clouds. *IEEE Trans. Autom. Sci. Eng.* **2020**, 1097–1106. [CrossRef]
6. Bi, J.; Yuan, H.; Zhou, M.C.; Liu, Q. Time-Dependent Cloud Workload Forecasting via Multi-Task Learning. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2401–2406. [CrossRef]
7. Yuan, H.; Bi, J.; Zhou, M.C. Temporal Task Scheduling of Multiple Delay-Constrained Applications in Green Hybrid Cloud. *IEEE Trans. Serv. Comput.* **2018**, 1. [CrossRef]
8. Jing, B.; Li, S.; Yuan, H.; Zhou, M. Integrated Deep Learning Method for Workload and Resource Prediction in Cloud Systems. *Neurocomputing* **2020**. [CrossRef]
9. Boss, G.; Malladi, P.; Quan, D.; Legregni, L.; Hall, H. Cloud Computing. *IBM White Pap.* **2007**, *1*, 1–17.
10. Schuster, F.; Costa, M.; Fournet, C.; Gkantsidis, C.; Peinado, M.; Mainar-Ruiz, G.; Russinovich, M. VC3: Trustworthy data analytics in the cloud using SGX. In Proceedings of the IEEE Symposium on Security and Privacy, San Jose, CA, USA, 17–21 May 2015.
11. Hashizume, K.; Rosado, D.G.; Fernández-Medina, E.; Fernandez, E.B. An analysis of security issues for cloud computing. *J. Internet Serv. Appl.* **2013**, *4*, 5. [CrossRef]
12. Arghandeh, R.; von Meier, A.; Mehrmanesh, L.; Mili, L. On the definition of cyber-physical resilience in power systems. *Renew. Sustain. Energy Rev.* **2016**, *58*, 1060–1069. [CrossRef]
13. Mell, P.; Grance, T. The NIST definition of cloud computing. In *Cloud Computing and Government: Background, Benefits, Risks*; Nova Science Publishers, Inc.: Hauppauge, NY, USA, 2011; ISBN 9781617617843.
14. Gonzalez, N.; Miers, C.; Redígolo, F.; Simplício, M.; Carvalho, T.; Näslund, M.; Pourzandi, M. A quantitative analysis of current security concerns and solutions for cloud computing. *J. Cloud Comput.* **2012**, *1*, 11. [CrossRef]
15. Zissis, D.; Lekkas, D. Addressing cloud computing security issues. *Future Gener. Comput. Syst.* **2012**, *28*, 583–592. [CrossRef]
16. Agrawal, T.; Singh, S.K. Analysis of security algorithms in cloud computing. In Proceedings of the 10th INDIACom 2016 3rd International Conference on Computing for Sustainable Global Development, INDIACom 2016, New Delhi, India, 16–18 March 2016.
17. Naone, E. Technology Overview, Conjuring Clouds. *MIT Technol. Rev.* **2009**, *1*, 54–56.
18. Anantwar, R.G.; Chatur, P.N.; Anantwar, S.G. Cloud Computing and Security Models: A Survey. *Int. J. Eng. Sci. Innov. Technol.* **2012**, *1*, 39–44.
19. Castell, S. Code of practice and management guidelines for trusted third party services. *INFOSEC Proj. Rep. S 2101* **1993**, *2*, 1–20.
20. Brodkin, B.J. Gartner: Seven cloud-computing security risks. *InfoWorld* **2008**, *1*, 1–3.
21. Alliance, C. Security research alliance to promote network security. *Netw. Secur.* **1999**, *1999*, 3–4. [CrossRef]
22. Marshall, D. Understanding Full Virtualization, ParaVirtualization, and Hardware Assist. *VMWare White Pap.* **2007**, *1*, 725–731.
23. Jericho Forum Cloud Cube Model: Selecting Cloud Formations for Secure Collaboration Problem. Available online: https://docplayer.net/2025431-Cloud-cube-model-selecting-cloud-formations-for-secure-collaboration.html (accessed on 19 July 2020).
24. Amazon. Available online: https://www.amazon.com/ (accessed on 19 July 2020).
25. Ramgovind, S.; Eloff, M.M.; Smith, E. The management of security in cloud computing. In Proceedings of the 2010 Information Security for South Africa Conference, ISSA 2010, Sandton, South Africa, 2–4 August 2010.
26. Suresh, K.S.; Prasad, K.V. Security Issues and Security Algorithms in Cloud Computing. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **2012**, *2*, 1–5.
27. Yuan, H.; Bi, J.; Zhou, M.; Sedraoui, K. WARM: Workload-Aware Multi-Application Task Scheduling for Revenue Maximization in SDN-Based Cloud Data Center. *IEEE Access* **2018**, *6*, 645–657. [CrossRef]
28. Yuan, H.; Bi, J.; Zhou, M.; Ammari, A.C. Time-Aware Multi-Application Task Scheduling with Guaranteed Delay Constraints in Green Data Center. *IEEE Trans. Autom. Sci. Eng.* **2018**, *15*, 1138–1151. [CrossRef]
29. Stanff, R.D. Securing the cloud with homomorphic encryption. *Next Wave* **2014**, *20*, 1–4.
30. Samavarchi, H.; Dehghani, M.R.; Ghasemzadeh, H. Improving Basic Cases of Data Mining. In Proceedings of the Computer and Network Technology-The International Conference on ICCNT 2009, Chennai, India, 24–26 July 2009.