International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 2 | ISSN : 2456-3307

Generate a Key to Secure a Personal Health Record in Medical Field

P. Parthasarathi, Gokulkumar R, Praveenkumar J, Ranganayaki L, Veena K

Department of Computer Science and Engineering, Akshaya College of Engineering and Technology, Coimbatore, Tamil Nadu, India

ABSTRACT

Information centric applications like remote healthcare application program are configured on a WSN-Cloud calculating incorporate architecture demands attention to its dependable information management mechanism. A Large quantity of information is rendered in a health care application program, which involves to be managed efficiently so that fast and dependable data communication and approach can come about among the end users. More and more IT companies are shifting to cloud based service like Private, Public and Hybrid cloud computing. But at the same time they are concerned about security issues. Accordingly, the demand for cloud computing has forced the development of new market offerings, representing various cloud service and delivery models. To enable secure and flexible data sharing in the cloud, efficient management of encryption keys is required. A key challenge to designing such encryption schemes lies in the efficient management of encryption keys. Updating the key without exchanging the key message will decrease the probability of accessing the key in other devices. The proposed system addresses the disease prediction operation which enables the way to access the large amount of data over the cloud network and the security methodology can be distributed a single constant-size aggregate key to a data user for sharing a large number of documents. Key-aggregate cryptosystem develop perpetual size ciphertexts such that effective process of decryption access rights for any set of ciphertext are possible. Any set of secret keys can be combined and make them as single key, which covers power of all the keys being combined. This combined key can be committed to the others for decryption of ciphertext set and persisting encrypted files outside the set are remains classified.

Keywords : Key-Aggregate Cryptosystem, WSN-Cloud, Public And Hybrid Cloud Computing, Decryption, Saas, Cloud Computing

I. INTRODUCTION

Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS), so we use that term. The datacenter hardware and software is what we will call a Cloud. In cloud computing, cloud service providers offer an abstraction of infinite storage space for clients to host data. To preserve data privacy, a common approach is to encrypt data files before the clients upload the encrypted data into the cloud. But the current cloud setup does posses the data leakage in the data sharing process. Cloud computing provides seemingly unlimited "virtualized" resources to users as services across the whole Internet, while hiding platform and implementation details. Today's cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. As cloud computing becomes prevalent, an increasing amount of data is being stored in the cloud and shared by users with specified privileges, which define the access rights of the stored data. One critical challenge of cloud storage services is the management of the ever-increasing volume of data.

In a cloud computing environment, many entities are involved. We are very much interested particularly in the cloud provider and the cloud consumer. The cloud provider is the entity which owns and manages the resources. The cloud consumer is the entity that consumes the resources and may be an individual or an organization. When a cloud consumer is an organization, it will have users or employees that access cloud resources. There may be casual users that have no relationship with the cloud provider, but are accessing cloud consumers' web services that are developed and hosted within the cloud. Here the security and trust relationship of concern is between the cloud provider and the cloud consumer.

Cloud computing promises several attractive benefits for businesses and end users. Cloud computing provides seemingly unlimited "virtualized" resources to users as services across the whole Internet, while hiding platform and implementation details. Today's cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. As cloud computing becomes prevalent, an increasing amount of data is being stored in the cloud and shared by users with specified *privileges*, which define the access rights of the stored data. One critical challenge of cloud storage services is the management of the ever-increasing volume of data.

In a cloud computing environment many entities are involved. We are very much interested particularly in the cloud provider and the cloud consumer. The cloud provider is the entity which owns and manages the resources. The cloud consumer is the entity that consumes the resources and may be an individual or an organization. When a cloud consumer is an organization, it will have users or employees that access cloud resources. There may be casual users that have no relationship with the cloud provider, but are accessing cloud consumers' web services that are developed and hosted within the cloud. Here the security and trust relationship of concern is between the cloud provider and the cloud consumer.

An alternative is the utilization of simulations tools that open the possibility of evaluating the hypothesis prior to software development in an environment where one can reproduce tests. Specifically in the case of Cloud computing, where access to the infrastructure incurs payments in real currency, simulation-based approaches offer significant benefits, as it allows Cloud customers to test their services in repeatable and controllable environment free of cost, and to tune the performance bottlenecks before deploying on real Clouds. At the provider side, simulation environments allow evaluation of different kinds of resource leasing scenarios under varying load and pricing distributions. Such studies could aid the providers in optimizing the resource access cost with focus on improving profits. In the absence of such simulation platforms, Cloud customers and providers have to rely either on theoretical and imprecise evaluations, or on try-and-error approaches that lead to inefficient service performance and revenue generation.

Security concerns relate to risk areas such as external data storage, dependency on the "public" internet, lack of control, multi-tenancy and integration with internal security. Compared to traditional technologies, the cloud has many specific features, such as its large scale and the fact that resources belonging to cloud providers are completely distributed, heterogeneous and totally virtualized. Traditional security mechanisms such as identity, authentication, and authorization are no longer enough for clouds in their current form. Security controls in Cloud Computing are, for the most part, no different than security controls in any IT environment. However, because of the cloud service models employed, the operational models, and the technologies used to enable cloud services, Cloud Computing may present different risks to an organization than traditional IT solutions. Unfortunately, integrating security into these solutions is often perceived as making them more rigid. Moving critical applications and sensitive data to public cloud environments is of great concern for those corporations that are moving beyond their data center's network under their control. To alleviate these concerns, a cloud solution provider must ensure that customers will continue to have the same security and privacy controls over their applications and services, provide evidence to customers that their organization are secure and they can meet their service-level agreements, and that they can prove compliance to auditors.

Convergent encryption has been proposed to enforce data confidentiality while making deduplication feasible. It encrypts/ decrypts a data copy with a *convergent key*, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the ciphertext to the cloud. Since the encryption operation is deterministic and is derived from the data content, identical data copies will generate the same convergent key and hence the same ciphertext. To prevent unauthorized access, a secure proof of ownership protocol is also needed to provide the proof that the user indeed owns the same file when a duplicate is found. After the proof, subsequent users with the same file will be provided a pointer from the server without needing to upload the same file. A user can download the encrypted file with the pointer from the server, which can only be decrypted by the corresponding data owners with their convergent keys.

A private cloud is a type of cloud computing that delivers similar advantages to the public cloud, including scalability and self-service, but through a proprietary architecture. Unlike public clouds, which deliver services to multiple organizations, a private cloud is dedicated to a single organization. As a result, aprivate cloud is best for businesses with dynamic or unpredictable computing needs that require direct control over their environments.

Public and private cloud deployment models differ. Public clouds, such as those from Amazon Web Services or Google Compute Engine, share a computing infrastructure across different users, business units or businesses. However, these shared computing environments aren't suitable for all businesses, such as those with mission-critical workloads, security concerns, uptime requirements or management demands. Instead, these businesses can provision a portion of their existing data center as an on-premises -- or private -- cloud.

But private clouds have some disadvantages. For example, on-premises IT -- rather than a third-party cloud provider -- is responsible for managing the private cloud. As a result, private cloud deployments carry the same staffing, management, maintenance and capital expenses as traditional data center ownership. Additional private cloud expenses include virtualization, cloud software and cloud management tools.

Streaming media is multimedia that is constantly received by and presented to an end-user while being delivered by a provider. A client end-user can use their media player to begin to play the data file before the entire file has been transmitted. Distinguishing delivery method from the media distributed applies specifically to telecommunications networks, as most of the delivery systems are either inherently streaming or inherently non-streaming. Live streaming refers to Internet content delivered in real-time, as events happen, much as live television broadcasts its contents over the airwaves via a television signal. Unicast protocols send a separate copy of the media stream from the server to each recipient. Unicast is the norm for most Internet connections, but does not scale well when many users want to view the same television program concurrently.

Multicast protocols were developed to reduce the server/network loads resulting from duplicate data streams that occur when many recipients receive unicast content streams independently. These protocols send a single stream from the source to a group of recipients. Depending on the network infrastructure and type, multicast transmission may or may not be feasible. HTTP adaptive bitrate streaming is based on HTTP progressive download, but contrary to the previous approach, here the files are very small, so that they can be compared to the streaming of packets, much like the case of using RTSP and RTP. A Telecoms network constitutes a accumulation of terminal clients, connections are associated so as to enable telecom between the terminals. The transmission system links associate the clients together. The nodes use circuit exchanging, message switching or packet switching to pass the signal through the correct links and clients to accomplish the correct destination terminal.

Internet network: access to the network allows users to use many resources. Over time the Internet network will replace books. This will enable users to discover information almost instantly and apply concepts to different situations. The Internet can be used for recreational, governmental, educational, and other purposes. Businesses in particular use the Internet network for research or to service customers and clients. The data network is used extensively throughout the world to connect individuals and organizations. Data networks can be connected to allow users seamless access to resources that are hosted outside of the particular provider they are connected to. The Internet is the best example of many data networks from different organizations all operating under a single address space.

Content Delivery Networks (CDNs) with built-in large-scale, distributed content caching mechanisms have been adopted in the Internet. CDNs are typically deployed and operated independently by thirdparty

CDN carriers. In both cases, content caching is one of the key mechanisms that make CDNs successful. However, content caching is only deployed as an overlay service rather than an inherent network capability, due to lack of the storage capability at individual routers.

In such networks, users focus only on contents, rather than the physical locations from which contents can be retrieved. Moreover, the network routing and in-network storage are most likely provisioned by the same network carrier in a content-centric manner. Hence, the cost of coordinating content caching and the performance of routing traffic become main concerns in CCN. On the other hand, non-coordinated caching mechanisms store only the locally most popular contents at each CCN router, without coordination with other routers. Therefore, such mechanisms not only incur less coordination cost but also are more likely to store less distinct contents due to lack of coordination. Furthermore, studies have shown that the popularity of both Web and video contents follows the Zipf distribution.

Therefore, there are clear trade-offs between the network performance and the coordination cost when designing innetwork caching mechanisms for CCN. More specifically, coordinated caching mechanisms may trade the coordination cost for the network performance, while non-coordinated caching mechanisms may incur a significantly lower cost on provisioning in-network caching and may degrade the network performance due to lack of fine-grained control on where contents are cached, retrieved and routed to users.

The streaming secured cloud is to be designed with the following objectives:

- To develop a simple holistic model to capture the network performance of routing contents to clients and the network cost incurred by globally coordinating the provision of the in-network storage capability.
- To derive the optimal strategy of provisioning the in-network storage capability to optimizes the overall network performance and cost, with mild conditions under which the optimal strategy is guaranteed to be unique.
- Through numerical analysis, to observe interesting phenomena. In particular, observing that the stability of the optimal strategy is sensitive to key factors such as the parameter of the content popularity distribution and the trade-off weights for the network performance and the coordination cost.

II. METHODS AND MATERIAL

EXISTING SYSTEM

Nowadays in so many hospitals they are using computerized technologies. so they are maintaining the person's health records in a server based systems. In server side we need secure communication between the person's health record server and authorized users but it is not happening nowadays because of key encryption are not well structured. To overcome that problem we are going to generate a well structured private key using key generation algorithms for securing the communication.

The system is based on multi-key searchable encryption scheme and key-aggregate cryptosystem. To enhance security and flexibility in data sharing, each PHR category is encrypted with different public keys. So in such scenario using traditional approach patient has to send all the encryption keys (K1, K2, ... Kp) to different data users likedoctors, nurses, insurance policy broker etc. depending on their access right.

Suppose Patient wants to share p documents with the doctor. In this scenario, Patient has to send all the encryption keys {ki} to the doctor and when doctor wants to retrieve records containing a keyword w, he has to generate Trapdoori for every record information encrypted with encryption key ki and submit all the trapdoors to the cloud server for query search.

The key idea of the Efficient Key Management Infrastructure (EKMI) is to divide the system into two domains namely public domain (PUDs) and personal domain (PSDs). The PUDs consists of user who make access based on their professional roles, such as doctors, nurses and medical researcher. The PSD users are personally associated with a data owner such as family members or close friends and they make access to Personal Health Records based on access rights assigned by the owner.

EKMI uses a new Decentralized Key Policy Attribute based Encryption (DKPABE) with user revocation in

Private Domain and Multi Authority Cipher text Policy Attribute Based Encryption (MACPABE) with attribute revocation in Public Domain. In revocation, Lazy revocation concept is used. The System proves to be Collusion Resistance (CR) by employing Tokenization concept in above algorithms.

The proposed EKMI system architecture is shown. The system has users like PHR owner, who stores the health information, users, who access the health information of owners based on access policy and administrators (admin) of various institutions. When the user registers with their personal details and password, each user is provided with a Global Identifier (GID), which can uniquely identify the user in the system. The user can login using the GID and the password through which the user is authenticated. The authenticated user can send request to owner, if he is personally related, to view the information. If the user is not personally related, he can gain access to the PHR based on the roles the user plays in an institution. These users come under Public domain. The user who wish to view the PHR of owner, request the data. If the access structure of the user satisfies the access policy, the Attribute Authority (AA) generates the key by replacing the GID with token from Tokenization server. Thus the generated private key and the encrypted data areshown to the user as they provide the key password. The data is then decrypted using the private key and displayed.

They are Private domain and Public domain. In Private domain Decentralized Key Policy Attribute Based Encryption used and in Public domain Multi-Authority Cipher text Policy Attribute Based Encryption is used. Tokenization is a process by which a sensitive data field is replaced with a surrogate value called token, which has no extrinsic or exploitable meaning or value. The token is a reference that maps back to the sensitive data through a tokenization system. The mapping from original data to a token uses methods which render tokens infeasible to reverse in the absence of the tokenization system, for example using tokens created from random numbers.

When token replace live data in systems, the result is minimized exposure of sensitive data to those applications, stores, people and processes, reducing risk of compromise or accidental exposure and unauthorized access to sensitive data. Tokenization systems may be operated in-house within a secure isolated segment of the data center, or as a service from a secure service provider. In EKMI system, we consider Global Identifier of a user to be sensitive data, as it has the power to throw lights on the access policy of the user and how the keys are correlated with the policy, which can ultimately lead to key forgery.

Revocation is the process of revoking the users from their access. It is achieved by updating the set of identity attributes in the access structure thereby updating the secret keys for non- revoked users at Cloud Service Provider (CSP). The data owner should maintain the membership list and the lifetime of the user. Once a user revocation event occurs, Cloud Servers just record information submitted by the data owner. If there is a file data access request from a user, the cloud server checks with the System's Users List (SUL) and encrypt the requested files and update the user's secret key if and only if the user is not revoked. This process of updating the user's secret key only when data is requested is known as Lazy revocation.

The user, request the CSP for the data of owner. The CSP in turn checks whether the user is authorized one or not and if the user is listed in System's User List. If authorized and not revoked, then it sends re-encrypted cipher text to the user and updates the secret key. From the key store, the user will obtain the Secret key which is used to decrypt the cipher text. The EKMI system uses User Revocation in PSD, as it is easy for the data owner to maintain the list of users who are related to them personally and to revoke the user as they wish. In PUD, Attribute Revocation is carried out based on the role played by the user and is carried by the administrator of the institution. User revocation is the process of removing the users from their access to the data. User revocation is achieved by updating System User's list by removing user id and the set of identity attributes in the access structure.

The CSP removes the user id from the SUL and store the re-encryption key in AHL. When the user sends request to access the data, the CSP checks whether the user id is listed in SUL. If listed, then the CSP generates a new secret key and updates to user. If not listed, the user cannot access the data. The CSP removes the user id from the SUL and store the re-encryption key in AHL. When the user sends request to access the data, the CSP checks whether the user id is listed in SUL. If listed, then the CSP generates a new secret key and update to user. If not listed, the user cannot access the data.

On receiving the request verify each owner's public key and the secret key for the attribute and decrypt each with help of secret key. The revoked users cannot decrypt ciphertext encrypted with public keys which is backward secrecy and new users who as enough attribute should also be able to decrypt previous ciphertext which are already encrypted with public keys previously. A PHR owner can specify the access privilege of data reader in their PSD, and let her application generate and distribute corresponding key to the latter. A reader in PSD could obtain the secret key by sending a request to the PHR owner via healthcare social-network (HSN), and the owner will grant her a subset of requested data types.

Another layer of symmetric encryption technique AES is also applied to the data before transmission. The key is derived at both ends during run time. Basically, an asymmetric technique is adopted in this stage to generate a new symmetric key. This method incorporates the advantages of both the symmetric and asymmetric techniques. The key distribution is done in asymmetric fashion which helps in signing the data whereas the encryption process with symmetric key works faster. Each time, before transmission, one public and one private key pair is generated. Using these values, at each end, one ephemeral shared secret is generated, which is used to derive the symmetric key. This key is further used to encrypt the patient record. This asymmetric key generation technique also helps in signing the data using the generated private part. As already stated, the remote health care application consists of a kiosk where the health personnel assists the patient. The patient details are collected at a particular kiosk and then these are sent to a doctor. Most of the time the health personnel (assumed here to be an trusted entity) take the decision to assign a particular doctor to a patient from the list of available doctors.

For any sort of data transmission between two entities, a secure channel has to be set up first. Whenever data needs to be exchanged, one symmetric key is generated at both end. For data transmission between two entities (e.g., between kiosk and cloud storage provider) encrypted data exchange is very important in the cloud environment. However, this wireless medium is also vulnerable to various attacks like MITM, DoS etc. Therefore, a hash value of the information can also be included with each data transmission. At the receiver end, if the hash value does not match with the transmitted data, then the key is supposed be compromised, hence new key generation request is raised.

User login and authentication information [ID, H(password)] are validated before giving the user access to any resource. This process has a number of functionalities like identity checking, role checking, role – key share. The identity provider will validate the user credential. On success, the corresponding user is provided with the proper access key, so that the user could retrieve the data is intended for.

User sends encrypted query about the patient data to the cloud storage provider (CSP) using its symmetric key. At the server side, the CSP decrypts the query and searches the metadata information. Upon retrieval of the requested patient's data, the decrypted data (decryption using storage key) is further encrypted using the user's symmetric key. The users are provided with the access keys depending on their role, on demand. Therefore, the intended data could only be visible at the user's end. This module is also provisioned for data to be only read only or not. This restriction is controlled from the interface level.

Drawbacks

1. The system can't able to process the large volume of file data
2. It provides the low level security mechanism due to complexity of data handling.
3. It does not provides the information about the medical data handling and its KAC fails to achieve the expected integrity.

LITERATURE REVIEW

The three main types of cloud computing has been studied and point out in the following sub sections. This subsection gives us a clear idea about different types of cloud computing.

Public Clouds

One of the standard cloud computing models is a Public cloud. Here the service provider makes resources like application, infrastructure and storage, available to the customers and businesses over the internet. The service providers like Microsoft, Google etc. have their own infrastructure at their data center. The access will be only through internet mode. There will be no direct connectivity is proposed in Public cloud architecture.The services may be offered free (Gmail) or may be provided as pay per use facilities to businesses. Here, the customer has limited visibility into or control over where the computing infrastructure is hosted. Although public cloud services are easy to administer and cost effective, they are not considered as secure as private clouds.

PrivateClouds

It is a cloud computing platform built on your own hardware and software. It is also known as internal cloud or corporate cloud. It provides hosted services to a limited number of people behind a firewall. When you require greater level of security and control over your applications, this type of cloud is most preferable. Here, the services and infrastructure provided are maintained over a private network and are generally used by corporate houses. This private cloud services is more costly because we need to buy, build and manage them. However, the reliability offered makes them popular and given them potential as a growing market.

Hybrid Clouds

When you wish to maintain different business applications with different levels of securityparticularly this service is useful. Hybrid clouds services are a combination of public and private clouds implemented by different providers. One of the disadvantages of these services is that we have to manage different security platforms together. This paper is mostly related to works in cryptographically enforced access control for outsourced data and attribute based encryption. To realize fine-grained access control, the traditional public key encryption (PKE)-based schemes , either incur high key management overhead, or require encrypting multiple copies of a file using different users' keys. To improve upon the scalability of the above solutions, one-to-many encryption methods such as ABE can be used. In Goyal et al.'s seminal paper on ABE , data are encrypted under a set of attributes so that multiple users who possess proper keys can decrypt. This potentially makes encryption and key management more efficient . A fundamental property of ABE is preventing against user collusion. In addition, the encryptor is not required to know the ACL.

Recently, Yu et al. (YWRL) applied key-policy ABE to secure outsourced data in the cloud , , where a single data owner can encrypt her data and share with multiple authorized users, by distributing keys to them that contain attribute-based access privileges. They also propose a method for the data owner to revoke a user efficiently by delegating the updates of affected ciphertexts and user secret keys to the cloud server. Since the key update operations can be aggregated over time, their scheme achieves low amortized overhead. However, in the YWRL scheme, the data owner is also a TA at the same time. It would be inefficient to be applied to a PHR system with multiple data owners and users, because then each user would receive many keys from multiple owners, even if the keys contain the same sets of attributes. On the other hand, Chase and Chow proposed a multiple-authority ABE (CC MA-ABE) solution in which multiple TAs, each governing a different subset of the system's users' attributes, generate user secret keys collectively. A user needs to obtain one part of her key from each TA. This scheme prevents against collusion among at most N _ 2 TAs, in addition to user collusion resistance. However, it is not clear how to realize efficient user revocation. In addition, since CC MA-ABE embeds the access policy in users' keys rather than the ciphertext, a direct application of it to a PHR system is nonintuitive, as it is not clear how to allow data owners to specify their file access policies.

In PaaS, developers do not usually have access to the underlying layers, so providers are responsible for securing the underlying infrastructure as well as the applications services. Even when developers are in control of the security of their applications, they do not have the assurance that the development environment tools provided by a PaaS provider are secure. In conclusion, there is less material in the literature about security issues in PaaS. SaaS provides software delivered over the web while PaaS offers development tools to create SaaS applications. However, both of them mayuse multi-tenant architecture so multiple concurrent users utilize the same software. Also, PaaS applications and user's data are also stored in cloud servers which can be a security concern as discussed on the previous section. In both SaaS and PaaS, data is associated with an application running in the cloud. The security of this data while it is being processed, transferred, and stored depends on the provider.

Razan Al-Ani et al (2011) have proposed a simulated and evaluated the AODV, OLSR, GRP, TORA and DSR routing protocols to analysis the performance on basis of Throughput, Delay, and Network load. They created a network which consists of mobile nodes, one fixed WLAN server running GRP and RX group config node to speed up simulation time. All nodes are configured to work with 5.5 Mbps data rate and FTP application type was chosen for all nodes with multiple FTP sessions. They ran four scenarios for each type of routing protocol. Each scenario was run for 30 min. According to their results OLSR routing protocol performs better than others in both delay and throughput.

Distributed End-To-End Allocation of Time Slots for Real-Time Traffic (DARE) have addressed by Emma Carlson et.al (2013). DARE is a channel access protocol for MANETs that provides end-to-end reservations. It is based on request-to-reserve messages that travel from source to destinations through routes established by a traditional on-demand routing protocol. Destinations replies with clear-to-reserve messages that travel along re-verse paths establishing the actual reservations. Data packets also contain reservation information and are used to refresh the reservations tables. The main limitations of DARE are that reservations are established at each hop of a path independently of the other hops in the path, and routing decisions do not consider information regarding reservations or any other data collected for channel access.

In-network caching necessitates the transformation of centralised operations of traditional, overlay caching techniques to a decentralised and uncoordinated environment. Given that caching capacity in routers is relatively small in comparison to the amount of forwarded content, a key aspect is balanced distribution of content among the available caches. In this paper, we are concerned with decentralised, real-time distribution of content in router caches. Our goal is to reduce caching redundancy and in turn, make more efficient utilisation of available cache resources along a delivery path. The in-network caching scheme, called ProbCache, approximates the caching capability of a path and caches contents probabilistically in order to: i) leave caching space for other flows sharing (part of) the same path, and ii) fairly multiplex contents of different flows among caches of a shared path.

Although the volume of Web traffic on the Internet is staggering, a large percentage of that traffic is redundant-multiple users at any given site request much of the same content. This means that a significant percentage of the WAN infrastructure carries the identical content (and identical requests for it) day after day. Eliminating a significant amount of recurring telecommunications charges offers an enormous savings opportunity for enterprise and service provider customers. Web caching performs the local storage of Web content to serve these redundant user requests more quickly, without sending the requests and the resulting content over the WAN.

Browser Freshness Controls

Finally, clients can always explicitly refresh content at any time by using the browser's reload/refresh button. The reload/refresh command is a browser-triggered command to request a data refresh. A reload/refresh will issue a series of IMS requests asking for only data that has changed. The shift+reload/shift+refresh command is an extension of the reload/refresh command. In correctly implemented browsers, this command always triggers a "pragma: no cache" rather than an IMS request. As a result, cache engines are bypassed and the end server directly fulfills all content.

Reverse Proxy Caching Function

Reverse Proxy Caching, each cache engine homes to WCCP-enabled routers/switches that are supporting server farms. When an incoming Web request reaches an WCCP-enabled router, the router performs a hashing function on the incoming request's source IP address and port number, mapping the request into one of 256 discrete buckets. Statistically, this hashing function distributes incoming requests evenly across all buckets. In addition, these buckets are evenly allocated among all cache engines in a cluster.

Because the hashing function is based on source IP address and port number instead of destination IP address, a given Web object could be stored in multiple cache engines in a cluster. By spreading popular content across a cache cluster, reverse proxy caching allows multiple cache engines to service requests for very popular content. Thus, additional cache engines can be added to a cluster to incrementally scale the performance of a popular site and decrease content download latency. Note that hashing on a destination IP address could also do the reverse-proxy caching. But in this case, all requests would have the same destination IP address and would be redirected to one cache engine. If you do not need to scale beyond one cache engine act as a front-end to a server farm, then this method is sufficient.

III. RESULTS AND DISCUSSION

PROPOSED SYSTEM

In cloud cryptography, the key aggregate policies is use to make a decryption of key more powerful in the sense that it allows decryption of multiple cipher texts, without increasing its size. A special type of public-key encryption which call key-aggregate cryptosystem. In KAC, users encrypt a message not only under a public-key, but also under an identifier of ciphertext called class. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for various classes. More importance, the extracted key have can be an aggregate key which is as compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of ciphertext classes. The sizes of ciphertext, public-key, and master-secret key and aggregate key in our KAC schemes are all of constant size. The public system parameter has size linear in the number of ciphertext classes, but only a small part of it is needed each time and it can be fetched oxmsn demand from large cloud storage

AUTHENTICATION AND KEY GENERATION

User is asked to keep this id as a secret because it is used as a tool to authenticate him every time he logs on to the system. Authentication merely ensures that the individual is who he (or) she pretends to be but says nothing about the access rights of the individual. Authenticated user has ID/password combinations to prove identity. Password authentication protocol, secure socket layer, digital signatures, Kerberos, firewall, virtual private networks are the techniques are used in authentication. Cipher class consists of Data owner’s id, message and the master/public key of the data owner attributes. Using the master key, public key is generated and secret key is generated by doing the logical XOR operations. Ciphering algorithms are applied using the secret key, thus secured secret key is generated by the Key aggregate cryptosystem.

The user must transmit a message of user authentication to the server, and then the server must be able to verify the identity of the user and give him the right of using permitted service. The user passes a password as a secret token to the server. First server checks the user identity if the password matches. The users identify the key distribution scheme with the ability of privacy protection, but it is pointed out that it is less efficient because of using public key cryptosystem. Some criteria basically on security and efficient requirements, we are using the user’s authentication and key management schemes with smart cards. The outcome will be criteria such as privacy protection, freely chosen password, low computation and communication cost, mutual authentication and session key agreements.

KEY AGGREGATION

Aggregate key is used for the secure data sharing over the distributed data sharing in cloud environment. Aggregate key consist of various derivation of identity and attribute based classes of respective data owner in the cloud. Aggregation key is used to sharing the data between one to other. The key aggregation property is especially useful when we expect the delegation to be efficient and flexible. Key aggregation enables content provider to share other’s data in a confidential and selective way, with a fixed and small cipher text expansion, by distributing to each authorized user a single and small aggregate key. Alice wants to share her data on the server. The key generation phase is provided by public key and master key pair. In this public and master key pairs are secretly done by Alice. Alice encrypts the data using public key and these data are uploaded to the server. Alice is willing to share a data to bob. Alice can compute the aggregate key for bob, it’s performed by master key, and this aggregate key is sent to bob via email and this aggregation key is used to download the data and decrypt the data. Extract is executed by the data owner for delegating the decrypting power for a certain set of cipher text classes to a delegate. In this example input is master key and data and output is aggregate key. It's the primary key having more than one column. Key aggregation is group of public key and private key used for transmission of data. The combination of public and private key is known as key aggregation. Key is nothing but composite or concatenated key. Example different books may have identical title, authors. In this case we can take title, author, and publication date as the aggregate key which acts as primary key. Map reduce function is also used in key aggregation. Advantage of key aggregation is such as a secure key cryptographic derivation, higher data security, supports data integrity process, and also easy to manage all the keys. For security issues key aggregation places a vital role to offer secured data transfer.

Merits of the Proposed System

1. The system can able to process the large volume of file data
2. Authentication and authorization model increases the security level of the proposed system.
3. It provides the information about the medical data handling and its Hybrid KAC achieves the expected integrity.

SYSTEM IMPLEMENTATION

The system has following implementation module.

1. Data collection and authentication
2. Patient information and Diagnosis process
3. Providing in-network services and creates connectivity with devices and Key generation setup
4. Key aggregation and data security.

Data collection and authentication

The main Administrator account in hospital management is used to create set of department, roles, possible diseases and the account is used to create the user enrollment process. Enrollment process for Doctor, Patient & CareGiver these is a unique id is generated for each login user. In the authentication process, the validation of user roles with Userid and password is executed and the successful authorization provides the access control for doctor, and it is provides the information about patient history and the provision for updating the diagnosis process.

On successful completion of caregiver login, it provides the provision for adding and viewing the information about the patient and their history. And also the updated information regarding the diagnosis process of the patient based on the input information such as symptoms. Based on the collected symptoms along with the doctor's input regarding the corresponding symptoms and its diagnosis information provided. The treatment information is extracted by applying the learning process by taking the symptoms and its diagnosis as input.

Personal information about the Patient is collected and the following fields, Patient id, Name, Address, Age, Height, Weight, and Gender. Patient Medical history is maintained in terms of review history, social history and medical history (treatment history) and it also contains the treatments provided by the caregiver or doctor. The disease handling form is used to apply the process of disease treatment in terms of the correlation-ship between patient and their collected history information. The process disease form is used to collect the symptoms about the various diseases and its treatment diagnosis. Diagnosis input is used to gather the information regarding the various diagnosis required corresponding to the symptoms of the disease.
Providing in-network services and creates connectivity

Setting up the network server to create the web interface and to create mobile interface for connection maintenance. Web service is created using the internet information service and the transceiver devices are registered with the network server using username and password. Server configuration algorithm is used setting up the in-network server in the existing system. The proposed system is designed by credentials verification algorithm with multi device recognition algorithm. The system takes the IP address of the server and device along with the Credentials of the user device as Input parameter. The system produces the Connection identifier generated for each user device.

Before initiating data access group user registration is performed to provide authorization. During the registration process an unique id is generated from system to every registered user. And an user entry is created in the cloud data base to establish authorized access to data

Key generation and key sharing

Once registration completes, the key generation system initiated to create keypair (pubkey,privkey) for each user. The cloud system is already initialized with system keypair. By combining these two keypair (user key, system key), a new shared key is generated to every user for data access. The shared key is stored in both user component and in cloud system database.

IBE KEY

Identity-based encryption is a public-key encryption in which the public-key of a user can be set as an identity-string of the user (e.g., an email address, mobile number). There is a private key generator (PKG) in IBE which holds a master-secret key, and issues a secret key to each user with respect to the user identity. The content provider can take the public parameter and a user identity to encrypt a message. The recipient can decrypt this CipherText by his secret key. In IBE scheme, key aggregation is constrained in the sense that all keys to be aggregated must come from different identity divisions‖.

While there are an exponential number of identities and thus secret keys, only a polynomial number of them can be aggregated. This significantly increases the costs of storing and transmitting Cipher Texts, which is impractical in many situations such as shared cloud storage. As Another way to do this is to apply the hash function to the string denoting the class, and keep hashing repeatedly until a prime is obtained as the output of the hash function.

Encryption decryption with data access

During data access, for both uploading and downloading process shared key is used to protect the data. In both paradigms, the sender encrypts the data using its public key which is derived from shared key. And the receiver decrypt the encrypted data using the private key derived from shared key after the decryption process completes, data integrity is ensured by using hash verification of data.

Key aggregation and delegation Data access is modeled in two types,

1. Set of user access the same data
2. Data access of single data file stored in multiple distributed database

In both access type either user key or database key is grouped together and an aggregated key is formed by delegating the corresponding user in the groups or delegating the databases for a file. the aggregated key is used to provide cloud data access for every user.

HYBRID ENCRYPTION STANDARD

To provide more security for data access in the cloud, identity and data based encryption method is designed. The initial user key is combined with identity information and IBE key is generated. IBE key is modeled with DBE system and hybrid key is generated for user. This hybrid key is shared with system key and new shared key is generated for user and system. The encryption and decryption process is done by using a new shared key which is generated by the hybrid system.

IV.CONCLUSION

A Large quantity of information is rendered in a health care application program, which involves to be managed efficiently so that fast and dependable data communication and approach can come about among the end users. But at the same time they are concerned about security issues. To enable secure and flexible data sharing in the cloud, efficient management of encryption keys is required. Updating the key without exchanging the key message will decrease the probability of accessing the key in other devices. The proposed system addresses the disease prediction operation which enables the way to access the large amount of data over the cloud network and the security methodology can be distributed a single constant-size aggregate key to a data user for sharing a large number of documents. Any set of secret keys can be combined and make them as single key, which covers power of all the keys being combined. This combined key can be committed to the others for decryption of CipherText set and persisting encrypted files outside the set are remains classified.

V. FUTURE WORK

In future by applying the generalization process, the designed medical environment with the large set of data regarding the disease's symptoms and diagnosis, the exact specification of the treatment can be identified by applying the supervised learning. With the generalized modification, the medical system can be used for urban and rural area at where the insufficient medical information is available.

VI. REFERENCES

[1]. Sascha Muller, Stefan Katzenbeisser, and Claudia Eckert, "On Multi- Authority Ciphertext-Policy Attribute-Based Encryption," Bull. Korean Math. Soc. 46 (2009), No. 4, pp. 803–819.

[2]. Vijayapriya M. and Malathi A, "Multi Authority Attribute Based Encryption for Personal Health Record in Cloud Computing‟, International Journal of Computer Trends and Technology (IJCTT), vol. 4, No. 8, 2013.

[3]. Nuttapong Attrapadung ,Benot Libert and Elie de Panafieu, "Expressive Key-Policy Attribute-Based Encryption with Constant-Size Cipher texts," 14 th International Conf. on Practice and Thoery in Public key Cryptography, vol. 6571, March 6-9, 2011, pp. 90-108.

[4]. M. Chase and S. S. Chow, "Improving privacy and security in multi authority attribute based encryption," in proceedings: ACM Conference on Computer and Communication Security – CCS'09(E. alshaer, S.Jha, and A.D. Keromytis, eds.), (Chinago, Illinois, USA), ACM, November 9-13, 2009, pp.121-130.

[5]. Reshma Mary Abraham and P. Sriramya, "Efficient and Secure Attribute Revocation of data in Multi-Authority Cloud Storage," APRN Journal of Engineering and Applied Sciences, Vol. 10, No. 13, July 2015, pp. 5588- 5592.

[6]. Ming Li, Shucheng Yu, and Yao Zheng, "Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption, " IEEE Transactions on Parallel and Distributed Systems, 24 (1), pp. 131-143, 2013.

[7]. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data, " Proc. 13th ACM Conf. Computer and Comm. Security (CCS 06), pp. 89-98, 2006.

[8]. M. Chase, and S.S.M. Chow, "Improving Privacy and Security in Multi- Authority Attribute-Based Encryption, " Proc. ACM Conf. Computer and Comm. Security, pp. 121-130. 2009.

[9]. R. Canetti, and S. Hohenberger, "Chosen-Ciphertext Secure Proxy Re- Encryption, " Proc. 14th ACM Conf. Computer and Comm. Security (CCS 07), pp. 185-194, 2007. 10C. K. Chu, and W. -G. Tzeng, "Identity-Based Proxy Re-encryption without Random Oracles, " Proc. Information Security Conf. (ISC 07), vol. 4779, pp. 189-202, 2007.

[10]. C. K. Chu, J. Weng, S.S.M. Chow, J. Zhou, and R.H. Deng, "Conditional Proxy Broadcast Re-Encryption, " Proc. 14th Australasian Conf. Information Security and Privacy (ACISP 09), vol. 5594, pp. 327-342, 2009.

[11]. S.S.M. Chow, J. Weng, Y. Yang, and R.H. Deng, "Efficient Unidirectional Proxy Re-Encryption, " Proc. Progress in Cryptology (AFRICACRYPT 10), vol. 6055, pp. 316-332, 2010.

[12]. J. W. Li, J. Li, and X. F. Chen, "Efficient Keyword Search over Encrypted Data with Fine-Grained Access Control in Hybrid Cloud, " In: Network and System Security 2012, LNCS, pp. 490-502, 2012.

[13]. Z. Liu, Z. Wang, and X. Cheng, "Multi-user Searchable Encryption with Coarser-Grained Access Control in Hybrid Cloud, " Fourth International Conference on Emerging Intelligent Data and Web Technologies (EIDWT), IEEE, pp. 249-255, 2013.

[14]. S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing, " Proc. IEEE INFOCOM, pp. 534-542, 2010.

[15]. F. Zhao, T. Nishide, and K. Sakurai, "Multi-User Keyword Search Scheme for Secure Data Sharing with Fine-Grained Access Control, " Information Security and Cryptology, LNCS, pp. 406-418, 2012.

[16]. D. Boneh, C. Gentry, and B. Waters, "Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys, " Proc. Advances in Cryptology Conf. (CRYPTO 05), vol. 3621, pp. 258-275,2005.

[17]. R. A. Popa ,and N. Zeldovich, "Multi-key searchable encryption, " Cryptology ePrint Archive, Report 2013/508, 2013.

[18]. J. Benaloh, "Key Compression and Its Application to Digital Fingerprinting," technical report, Microsoft Research, 2009.

[19]. T. Okamoto and K. Takashima, "Achieving Short Ciphertexts or Short Secret-Keys for Adaptively Secure General Inner-Product Encryption, " Proc. 10th Intl Conf. Cryptology and Network Security (CANS 11), pp. 138-159, 2011.

[20]. Department of Veterans Affairs, "VA Personal Health Record Non- Identifiable Data," http://catalog.data.gov/dataset/va-personal-health- record-non-identifiable-data (accessed September 8, 2014).