ISSN:0975-9646

Fine Grained Data Access Control in Cloud Computing

A.Parameshwari, B.Rasina Begum

Department of Computer Science and Engineering
Mohamed Sathak Engineering College, Kilakarai.

Abstract— **The uprising of medical field is distribution secure Personal Health Record (PHR) via the internet. Personal Health Record (PHR) is a health record where health data and information related to the care of a patient is kept by the patient. The PHR owner outsources the PHR to the third party servers for the widespread database management and for the security. The patient records should be maintained with high privacy and security. The security systems are used to protect the personal data from public access. Patient data can be retrieved by many different people. Each authority is assigned with access permission for a particular set of attributes. The access control and privacy management is a complex task in the patient health record management process. Data owners update the personal data into third party cloud data centers. This project proposes a novel patient-centric framework and a suite of data access mechanisms to control PHRs stored in semi-trusted servers. To achieve fine-grained and scalable data access control for PHRs, it leverages Attribute Based Encryption (ABE) techniques to encrypt each patient's PHR file. Multiple data owners can access the same data values. The proposed scheme could be extended to Multi Authority Attribute Based Encryption (MA-ABE) for multiple authority based access control mechanism.**

Keywords— **multiple authority, dynamic revocation, cloud computing, data privacy, fine-grained access control, attribute based encryption.**

I.INTRODUCTION

A personal health record (PHR) makes it easy to gather and manage medical Information in one accessible and secure location. Carrying paper records is a big drawback, rarely the patients have with them when they need. Personal health record systems overcome this problem by making the personal health record accessible anytime via a Web-enabled device, such as computer. Literally, having a personal health record can be a lifesaver. In an emergency patient can quickly give emergency personal vital information about disease, medications and drug allergies. Now a day, personal health record (PHR) has become a patient-centric model of health information exchange. A PHR service allows patients to create, manage and control their personal health data from one place through the web, which has made the storing, retrieving and sharing of the medical information more efficient. Especially, each patient will have full control of their medical records and can share their health data with different users from different domains which include healthcare providers, family members and friends. Due to high cost of building and maintaining separate data centers , many PHR services are outsourced and provided by third party service providers for example, Microsoft Health Vault (UK).

The Health Vault Program of Microsoft will allow users including individuals, health centers, hospitals etc. to gain access to the information on health related issues. The user interface will be simple, that would allow anyone to operate the program easily. But on the other hand, many people do not wish to share their private health records and other information universally through the Health Vault. As the sensitive Personal Health Information (PHI) is highly valuable, the third-party storage servers are the targets of various malicious behaviours which may result in exposure of the PHI. Researches on PHR's using cloud computing is still underway. The main concern is about whether the patients could actually control the sharing of their sensitive PHR and other information, especially when they are stored on a third-party server which people may not fully trust. To ensure patient-centric privacy control over their own PHRs, encryption of data is necessary prior storage. Basically, the PHR owners themselves should decide how to encrypt their files and to allow which set of users to obtain access to each file. The PHR and other files are available to only those users who are given the corresponding decryption key and are confidential to other users. Furthermore, the patient will always have the right to not only to grant, but also to revoke access rights when it is necessary. This scheme endeavours to study the patient-centric secure sharing of PHRs stored on semi-trusted servers, and focus on addressing the complicated and challenging key management issues.

II. RELATED WORK

1) Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute Based Encryption

Personal health record (PHR) is an emerging patient-centric model of health information exchange, which is often outsourced to be stored at a third party, such as cloud providers. However, there have been wide privacy concerns as personal health information could be exposed to those third party servers and to unauthorized parties. To assure the patients' control over access to their own PHRs, it is a promising method to encrypt the PHRs before outsourcing. Yet, issues such as risks of privacy exposure, scalability in key management, flexible access and efficient user revocation, have remained the most important challenges toward achieving fine-grained, cryptographically enforced data access control.

In this project, it proposes a novel patient-centric framework and a suite of mechanisms for data access control to PHRs stored in semi-trusted servers. To achieve fine-grained and scalable data access control for PHRs, it leverage attribute based encryption (ABE) techniques to encrypt each patient's PHR file. Different from previous works in secure data outsourcing, it focuses on the multiple data owner scenario, and divides the users in the PHR system into multiple security domains that greatly reduces the key management complexity for owners and users.

A high degree of patient privacy is guaranteed simultaneously by exploiting ABE. This scheme also enables dynamic modification of access policies or file attributes, supports efficient on-demand user/attribute revocation and break-glass access under emergency scenarios. Extensive analytical and experimental results are presented which show the security, scalability, and efficiency of the proposed scheme.

2) *Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing*

To keep sensitive user data confidential against untrusted servers, existing solutions usually apply cryptographic methods by disclosing data decryption keys only to authorized users. However, in doing so, these solutions inevitably introduce a heavy computation overhead on the data owner for key distribution and data management when fine-grained data access control is desired, and thus do not scale well. The problem of simultaneously achieving fine-grainedness, scalability, and data confidentiality of access control actually still remains unresolved. This paper addresses this challenging open issue by, on one hand, defining and enforcing access policies based on data attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in fine grained data access control to untrusted cloud servers without disclosing the underlying data contents.

Existing work can be found in the areas of shared cryptographic file systems and access control of outsourced data.

The Proposed system combining techniques of attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption. The proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability.

The main issue with this scheme is that collusion between a malicious server and any single malicious user would expose decryption keys of all the encrypted data and compromise data security of the system completely. In addition, user access privilege is not protected from the proxy server. User secret key accountability is neither supported.

3) *Authorized Private Keyword Search over Encrypted Personal Health Records in Cloud Computing*

In this paper, it formulates and addresses the problem of authorized private keyword searches (APKS) on encrypted PHR in cloud computing environments. It first present a scalable and fine-grained authorization framework for searching on encrypted PHR, where users obtain query capabilities from localized trusted authorities according to their attributes, which is highly scalable with the user scale of the system.

The existing solutions of searchable encryption are still far from practical for PHR applications in cloud computing environments. First and foremost, they are limited both in the type of applications and system scalability.

It proposes two novel solutions for APKS based on a recent cryptographic primitive, hierarchical predicate encryption (HPE), one with enhanced efficiency and the other with enhanced query privacy. In addition to document privacy and query privacy, other salient features of this scheme include efficiently support multi-dimensional, multiple keyword searches with simple range query; allow delegation and revocation of search capabilities.

4) *Improving Privacy and Security in Multi-Authority Attribute-Based Encryption*

Attribute based encryption (ABE) determines decryption ability based on a user's attributes. In a multi-authority ABE scheme, multiple attribute-authorities monitor different sets of attributes and issue corresponding decryption keys to users and encryptions can require that a user obtain keys for appropriate attributes from each authority before decrypting a message.

Existing system uses a fuzzy IBE scheme, which allowed for some error tolerance around the chosen identity. In more recent terminology, it would be described as a key-policy (KP) ABE scheme that allows for threshold policies. Key-policy means that the encryptor only gets to label a ciphertext with a set of attributes. The authority chooses a policy for each user that determines which ciphertexts he can decrypt. A threshold policy system would be one in which the authority specifies an attribute set for the user, and the user is allowed to decrypt whenever the overlap between this set and the set associated with a particular ciphertext is above a threshold.

The proposed system makes use of some basic techniques in anonymous credential systems to protect the privacy of ABE users. In an anonymous credential system, users wish to obtain and prove possession of credentials while remaining anonymous.

A multi-authority ABE system which requires a user to present his unique identifier to every authority would have severe privacy shortcomings.

5) *Shared and Searchable Encrypted Data for Untrusted Servers*

Current security mechanisms are not suitable for organizations that outsource their data management to untrusted servers. Encrypting and decrypting sensitive data at the client side is the normal approach in this situation but has high communication and computation overheads if only a subset of the data is required, for example, selecting records in a database table based on a keyword search. New cryptographic schemes have been proposed that support encrypted queries over encrypted data. But they all depend on a single set of secret keys, which implies single user access or sharing keys among multiple users, with key revocation requiring costly data re-encryption.

Existing schemes for searchable data encryption in multi-user settings which have constraints such as asymmetric user permissions (multiple writers, single reader) or read-only shared data set, in this scheme the shared data set can be updated by the users and each user in the group can be both reader and writer. The server can search on the encrypted data using encrypted keywords.

In this project, propose an encryption scheme where each authorized user in the system has own keys to encrypt and decrypt data. The scheme supports keyword search which enables the server to return only the encrypted data that satisfies an encrypted query without decrypting it.

Searchable encryption is difficult because searching leaks information about stored data inevitably. As long as the searching algorithm is correct, it always returns the same result set for the same query. Although the queries and the result sets are encrypted, the adversary can still build up search patterns.

III. PROBLEM STATEMENT

Generally PHR system has multiple PHR owners and PHR users. The owners refer to patients who have full control over their own PHR data, i.e., they can create, manage and delete it. There is a central server belonging to the PHR service provider that stores all the owners‟ PHRs. The users may come from various aspects; like, a friend, a caregiver or a researcher. Users access the PHR documents through the server in order to read or write to someone's PHR, and a user can simultaneously have access to multiple owners‟ data. Correctness of the PHI in the cloud is put at risk due to the following reasons. Although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they still face a broad range of both internal and external threats to data integrity.

Outsourcing data into the cloud is economically attractive for the cost and complexity of long-term large scale data storage, it does not offer any guarantee on data integrity and availability. This problem, if not properly addressed, may spoil the successful deployment of the cloud architecture. To fully ensure data security and save PHR owners computation resources, propose to the framework, where data owners can resort to an external third party auditor (TPA) to verify the outsourced data when needed. Third party auditing provides a transparent yet cost-effective method for establishing trust between PHR owner and cloud server. In fact , based on the audit result from a TPA, the released audit report would not only help PHR owners to evaluate the risk of their subscribed cloud data services, but also be beneficial for the cloud service provider to improve their cloud based service platform.

IV.MODELS, DESIGN GOALS AND ASSUMPTIONS

1) ***System Model***

The proposed system consists of Data Provider, Data Consumers and Cloud Service provider. Data providers use the storage capacity provided by CSP by uploading the encrypted files for exchange. Data consumers download a copy of data from cloud server and decrypt it by using his decryption key. Neither data provider nor the user is always online.CSP is always online and has storage capacity and computation power.

2) ***Security Model***

Always the server to be semi-trusted that is honest but curious as malicious access. That means the server will try to find out as much secret information in the stored PHR files as possible, but they will honestly follow the protocol in general. On the other hand, some users will also try to access the files beyond their privileges.

CSP to be semi trusted, i.e,"honest but curious". That means the cloud server will honestly perform the task delegated by the owner, but they will try to find out as much sensitive information in stored medical data as possible. At the same time, some users will also try to access the files beyond their scope of access privileges. For e.g., Drug companies may want to obtain the prescriptions of patients for understanding the buying patterns and boosting their profits. To do so they may collude with cloud servers for getting beneficial results. The Proposed work focuses on fine grained access control in a cloud based medical data exchange

3) ***Design Goals***

Our main goal is to achieve secure patient centric medical access control and secure key management at same time. The system guarantees negligible execution overhead on both the owner and user, while allowing guaranteed user revocation. The proposed method should prevent cloud servers from knowing both data file contents and access privilege information of user.

V. EXISTING SYSTEM

In Existing system of a PHR system model, there are multiple owners who may encrypt according to their own ways, possibly using different sets of cryptographic keys. Letting each user obtain keys from every owner who's PHR she wants to read would limit the accessibility since patients are not always online. An alternative is to employ a central authority (CA) to do the key management on behalf of all PHR owners, but this requires too much trust on a single authority (i.e., cause the key escrow problem).Key escrow (also known as a **"**fair" cryptosystem) is an arrangement in which the keys needed to decrypt encrypted data are held in escrow so that, under certain circumstances, an authorized third party may gain access to those keys. These third parties may include businesses, who may want access to employees' private communications, or governments, who may wish to be able to view the contents of encrypted communications. For the secured sharing of personal health record, the data is stored in cloud server and the key management is provided by the single trusted authority based attribute-based encryption (ABE).

Using ABE policies are expressed based on the attributes of users or data, which enables a patient to selectively share her PHR among a set of users by encrypting the file under a set of attributes, without the need to know a complete list of users. The complexities per encryption, key generation and decryption are only linear with the num

Drawbacks

- It not only leads to load bottleneck, but also creates the key escrow problem.
- As it is a single trusted authority there may be user collision due to the confusion in key distribution.
- It is not secured to delegate the key management for all attributes to the single trusted authority.
- Key management scalability, dynamic policy updates, and efficient on-demand revocation are non-trivial to solve, and remain largely open up-to-date.

VI.Proposed System

First, the system is divided into multiple security domains like Personal domain (PSD) and Public domain (PUD). Each domain controls only a subset of its users. For each security domain, one or more authorities are assigned to govern the access of data. For personal domain it is the owner of the PHR itself who manages the record and performs key management. This is less laborious since the number of users in the personal domain is comparatively less and is personally connected to the owner.

On the other hand, public domain consists of a large number of professional users and therefore cannot be managed easily by the owner herself. Hence it puts forward the new set of public Attribute Authorities (AA) to govern disjoint subset of attributes distributively.

A Multi-Authority ABE system is comprised of attribute authorities and one central authority. In this framework, there are multiple SDs, multiple owners, multiple AAs, and multiple users. In addition two ABE systems are involved: for each PSD the YWRL's revocable KP-ABE scheme is adopted; for each PUD, the proposed revocable MA-ABE scheme. Each data owner (e.g., patient)is a trusted authority of their own PSD, who uses a KP-ABE system to manage the secret keys and access rights of users in their PSD .Secondly, so as to achieve security of health records, a new encryption pattern namely Attribute based encryption (ABE) is adopted.

Data is classified according to their attributes. In certain cases, users may also be classified accordingly into roles. PHR owner encrypts their record under a selected set of attributes and those users that satisfy those attributes can obtain decryption key in order to access the data.

However, in the new solution pattern, an advanced version of ABE called multi-authority ABE (MA-ABE) is used. In this encryption scheme, many attribute authorities operate simultaneously, each handing out secret keys for a different set of attributes.

Advantages

- It Avoids Key Escrow Problem.
- Maintain Better Security And Privacy.
- Complexity Of Key Management Greatly Reduced.
- Support Dynamic Policy Changes, Enforced Write Access Control.
- More Expressive File Access.
- Reduced Storage and Communication Cost.
- Enhances the System Scalability.

VII.Fine Grained Data Access Control

1) *Revocable ABE*

It is a well-known challenging problem to revoke users/attributes efficiently and on-demand in ABE. Traditionally this is often done by the authority broadcasting periodic key updates to unrevoked users frequently, which does not achieve complete backward/forward security and is less efficient. Recently proposed two CP-ABE schemes with immediate attribute revocation capability, instead of periodical revocation. However, they were not designed for MA-ABE. IN addition, Ruj et al proposed an alternative solution for the same problem in this paper using Lewko and Waters's (LW) decentralized ABE scheme. The main advantage of their solution is, each user can obtain secret keys from any subset of the TAs in the system, in contrast to the CC MA-ABE. The LW ABE scheme enjoys better policy expressiveness, and it is extended to support user revocation. On the downside, the communication overhead of key revocation is still high, as it requires a data owner to transmit an updated ciphertext component to every non-revoked user. They also do not differentiate personal and public domains. In this project, it bridges the above gaps by proposing unified security framework for patient-centric sharing of PHRs in a multi-domain, multi-authority PHR system with many users. The framework captures application level requirements of both public and personal use of a patient's PHRs, and distributes users' trust to multiple authorities that better reflects reality. It also proposes a suite of access control mechanisms by uniquely combining the technical strengths of both CC MA-ABE and the YWRL ABE scheme. Using our scheme, patients can choose and enforce their own access policy for each PHR file, and can revoke a user without involving high overhead. It also implements part of this solution in a prototype PHR system.

2*) Enhancing MA-ABE for user revocation*

The original CC MA-ABE scheme does not enable efficient and on-demand user revocation. To achieve this for MA-ABE, it combines ideas from YWRL's revocable ABE and propose an enhanced MA-ABE scheme. In particular, an authority can revoke a user or user's attributes immediately by re-encrypting the ciphertexts and updating users' secret keys, while a major part of these operations can be delegated to the server which enhances efficiency. The idea to revoke one attribute of a user in MA-ABE is as follows. The AA who governs this attribute actively updates that attribute for all the affected unrevoked users. To this end, the following updates should be carried out the public/master key components for the affected attribute; the secret key component corresponding to that attribute of each unrevoked user; also, the server shall update all the ciphertexts containing that attribute. In order to reduce the potential computational burden for the AAs, it adopts proxy encryption to delegate operations to the server, and use lazy revocation to reduce the overhead.

3) *Enforce Write Access Control*

If there is no restriction on write access, anyone may write to someone's PHR using only public keys, which is undesirable. By granting write access, it means a data contributor should obtain proper authorization from the organization she is in (and/or from the targeting owner), which shall be able to be verified by the server who grants/rejects write access.

A naive way is to let each contributor obtain a signature from her organization every time she intends to write. Yet this requires the organizations be always online. The observation is that, it is desirable and practical to authorize according to time periods whose granularity can be adjusted. For example, a doctor should be permitted to write only during her office hours; on the other hand, the doctor must not be able to write to patients that are not treated by her. Therefore, it combines signatures with the hash chain technique to achieve that goal.

4) *ABE for Fine-grained Data Access Control*

ABE to realize fine-grained access control for outsourced data especially; there has been an increasing interest in applying ABE to secure electronic healthcare records. An attribute-based infrastructure for EHR systems, where each patient's EHR files are encrypted using a broadcast variant of CP-ABE that allows direct revocation. However, the cipher text length grows linearly with the number of unrevoked users. In a variant of ABE that allows delegation of access rights is proposed for encrypted EHRs applied cipher text policy ABE (CP-ABE) to manage the sharing of PHRs, and introduced the concept of social/professional domains investigated using ABE to generate self-protecting EMRs, which can either be stored on cloud servers or cell phones so that EMR could be accessed when the health provider is offline.

5) *System Setup and Key Distribution*

The system first defines a common universe of data attributes shared by every PSD, such as "basic profile", "medical history", "allergies", and "prescriptions". An emergency attribute is also defined for break-glass access. Each PHR owner's client application generates its corresponding public/master keys.

The public keys can be published via user's profile in an online healthcare social-network (HSN). There are two ways for distributing secret keys. First, when first using the PHR service, a PHR owner can specify the access privilege of a data reader in her PSD, and let her application generate and distribute corresponding key to the latter, in a way resembling invitations in GoogleDoc. Second, a reader in PSD could obtain the secret key by sending a request (indicating which types of files she wants to access) to the PHR owner via HSN, and the owner will grant her a subset of requested data types.

6*) PHR Encryption and Access*

The owners upload ABE-encrypted PHR files to the server. Each owner's PHR file is encrypted both under a certain fine grained and role-based access policy for users from the PUD to access, and under a selected set of data attributes that allows access from users in the PSD. Only authorized users can decrypt the PHR files, excluding the server. For improving efficiency, the data attributes will include all the intermediate file types from a leaf node to the root. For example, in an "allergy" file's attributes are {PHR, medical history, allergy}. The data readers download PHR files from the server, and they can decrypt the files only if they have suitable attribute based keys. The data contributors will be granted write access to someone's PHR, if they present proper write keys.

7) *User Revocation*

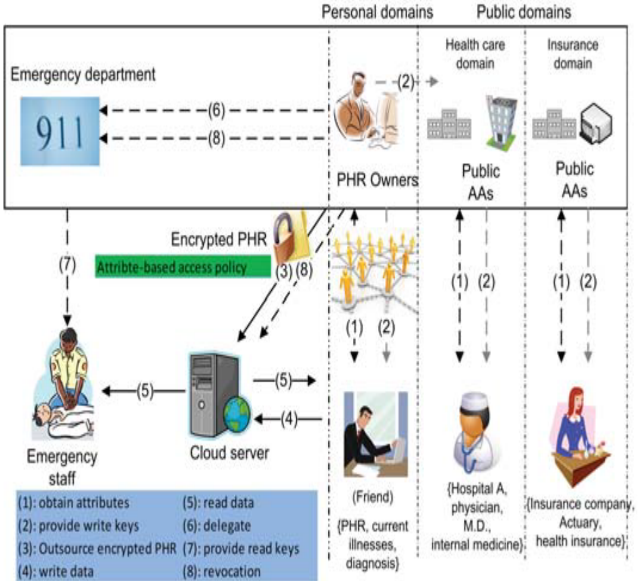
Here it considers revocation of a data reader or her attributes/access privileges. There are several possible cases revocation of one or more role attributes of a public domain user, revocation of a public domain user which is equivalent to revoking that entire user's attributes. These operations are done by the AA that the user belongs to, where the actual computations can be delegated to the server to improve efficiency. Revocation of a personal domain user's access privileges. These can be initiated through the PHR owner's client application in a similar way.

8) *Policy Updates*

A PHR owner can update her sharing policy for an existing PHR document by updating the attributes (or access policy) in the cipher text. The supported operations include add/delete/modify, which can be done by the server on behalf of the user.

9) Break-glass

When an emergency happens, the regular access policies may no longer be applicable. To handle this situation, break-glass access is needed to access the victim's PHR. In this framework, each owner's PHR's access right is also delegated to an emergency department ED. To prevent from abuse of break-glass option, the emergency staff needs to contact the ED to verify her identity and the emergency situation, and obtain temporary read keys. After the emergency is over, the patient can revoke the emergent access via the ED.



VIII.Conclusion And Future Work

This project proposes a novel framework of achieving fine grained access control for sharing personal data. Considering partially trustworthy cloud servers, it argues that to fully realize the patient-centric concept, patients shall have complete control of their own privacy through encrypting their PHR files to allow fine-grained access. The framework addresses the unique challenges brought by multiple PHR owners and users, in that greatly reduce the complexity of key management while enhance the privacy guarantees compared with previous works. It utilizes ABE to encrypt the PHR data, so that patients can allow access not only by personal users, but also various users from public domains with different professional roles, qualifications, and affiliations. Furthermore, enhance an existing MA-ABE scheme to handle efficient and on-demand user revocation, and prove its security.

As future study, it will be interesting to enhance the fine grained access control in cloud computing with a third party auditor to verify the cloud server that stores and process the PHRs. Homomorphic Split key Encryption can become additional enhancement to verify the trustworthiness of the TPA.

References

[1] M. Li, S. Yu, K. Ren, and W. Lou, "Securing Personal Health Records in Cloud Computing: Patient-Centric and Fine-Grained Data Access Control in Multi-Owner Settings," Proc. Sixth Int'lICST Conf. Security and Privacy in Comm. Networks (SecureComm '10), pp. 89-106, Sept. 2010.

[2] H. Lo¨ hr, A.-R.Sadeghi, and M. Winandy, "Securing the E-Health Cloud," Proc.First ACM Int'l Health Informatics Symp.(IHI '10),pp. 220-229, 2010.

[3] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized Private Keyword Search over Encrypted Personal Health Records in Cloud Computing," Proc. 31st Int'l Conf. Distributed Computing Systems(ICDCS '11), June 2011.

[4] "The HealthInsurance Portability and Accountability Act,"http://www.cms.hhs.gov/HIPAAGenInfo/01_Overview.asp,2012.

[5] "Google, Microsoft Say Hipaa Stimulus Rule Doesn't ApplytoThem," http://www.ihealthbeat.org/Articles/2009/4/8/, 2012.

[6] "At Risk of Exposure - in the Push for Electronic Medical Records,"http://articles.latimes.com/2006/jun/26/health/he-privacy26, 2006.

[7] K.D. Mandl, P. Szolovits, and I.S. Kohane, "Public Standards and Patients' Control: How to Keep Electronic Medical Records Accessible but Private," BMJ, vol. 322, no. 7281, pp. 283-287, Feb. 2001.

[8] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, "Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records," Proc. ACM Workshop Cloud Computing Security (CCSW '09), pp. 103-114, 2009.

[9] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing, "Proc. IEEE INFOCOM '10, 2010.

[10] C. Dong, G. Russello, and N. Dulay, "Shared and Searchable Encrypted Data for Untrusted Servers," J. Computer Security, vol. 19, pp. 367-397, 2010.