# Key Factor Authentication and Access Control for Accessing Cloud Computing Services

S. Eswari
Research Scholar,
Department of Computer Science
Mother Teresa Women's University
Kodaikannal

S. Manikandan, PhD
Profrssor and Head,
Dept of CS and Engineering,
Sriram Engineering College,
Perumalpattu, Chennai-602024

## ABSTRACT
In this paper, the new algorithm named Key Factor Authentication and Access Control (KFAAC) is used to ensure the security and access control of cloud services from the clients. Cloud and Web services becoming more popular in recent days. Instead of using offline applications peoples intended to use online applications and services with updated features with recent crisis. Providing best services to the client is becoming a major role for cloud service providers. In the same time better access restriction and authentication policy should be preserved. This proposed system Key Factor Authentication and Access Control provide better access security for the cloud clients. It uses Key Encryption Scheme to validate client to ensure their uniqueness. KFA is implemented and tested under cloud environment and it shows better access control and security.

## Keywords
Access Control, Web services, Cloud Security, DES encryption, privacy preserving authentication, Cloud Service security.

## 1. INTRODUCTION
Cloud computing is hot trending, commercially gaining and flexible environment for providing online services for the Client over the Internet. The Client can register on cloud and access the cloud services for free or by pay per service manner. In cloud environment there is no need for the physical system implementation web servers, database servers and file servers can be implemented virtually [2]. There are many cloud services like data sharing, data storage, database storage and online applications are provided by some service providers. The consumer who are the clients are provided with the client side application or online account to access their permitted services. Web browser, mobile application and desktop application are some client side access to the cloud. The consumers are allowed to access the web services remotely from anywhere via the Internet services. It is the responsibility of the cloud service providers to provide flexible, secured, efficient, accurate and timely services to the client.

Presently we are lies in the era of computer. Computer is ruling the world. It is everywhere where we go. Also consumers of cloud network are becoming high [1]. There are 3,424,971,237 of consumers using the Internet everyday around world. It is 46.1% of world population consuming Internet in their everyday use. So there are such a number of services accessed from the cloud everyday. The service providers has to preserve access control for the client access and should provide access only to those eligible users. A better access control and authentication scheme should be implement on the cloud network. A user should authenticate himself before accessing a service from the cloud [3]. The traditional way of authenticating client using username and password is no more secure due to the advanced technique of hackers like key logging and phishing pages attacking [22]. But it can also consider as one of the parameters of authentication [5]. Another method of recent day authentication is device based and application based authentication. The consumer is allowed to access cloud services only from the desired software application. The consumer has to authenticate himself from that software application as the first stage. Then the service provider will validate the software application by its ID. But there are many fake malicious applications are developed by the hackers to spoof the service providers and the clients. So the traditional attribute based authentication scheme is no more valid for recent malicious environment.

So there should be implement a better authentication scheme for the cloud environment to validate the consumer on the network. The proposed system should do the following things,

- It should authenticate the consumer to ensure consumer as valid client.
- It should validate consumer access to ensure consumer's access rights.
- It should provide flexible and flawless services to the consumer.

To solve those above problems this paper presents a Key Factor Authentication and Access Control (KFAAC). Instead of validating key from the consumer the KFAAC sends encrypted access key to the client application after validating the password authentication. And if the consumers application has the proper decryption algorithm to decrypt and use the access key send by the cloud, then it can able to access services from that could. This process is repeated for every new access of the services from the cloud. The proposed algorithm preserves consumer privacy and security [24]. This algorithm also provides flexible access to the consumers with flawless services. In addition to this the algorithm is capable to track consumer's access from device, application, location, operating system and network type to ensure the privacy and security.

The following section shows collection of related work relevant to our proposed scheme under cloud computing and other authentication schemes.

## 2. RELATED WORK

Several review work is done on different type of authentication scheme used on recent days. They are discussed one by one.

### 2.1 An Authenticated Trust and Reputation Calculation

It authenticates the Cloud Service Provider and Sensor Network Provider [1]. It also sets attribute requirements of the cloud service provider and cloud service user to access the web. It calculates the trust and reputation regarding the service of cloud service provider and Sensor Network Provider [25][28]. It helps the Cloud Service user to choose desirable Cloud Service Provider by assisting the Cloud Service Provider selecting Sensor Network Provider.

### 2.2 Smart Card Generator

This scheme includes three phases system setup, registration and authentication. The Smart Card Generator first setup the master private key by generating a random number. And then computes the corresponding public key and attributes. Then it publishes public key and public parameters. The consumer can utilize the public key for accessing purpose. The service provider and the consumer has to register with the SCG [2]. The identities were sent to the service provider and the consumer. These identities is later verified for the access control. The identity based cryptosystem is used to encrypt and decrypt the user data over the cloud.

### 2.3 Location-based arbitrary-subspace skyline queries

This paper concentrates on Location Based Service authentication. Merkle Skyline R-Tree is used to process the queries. Partial S4-Tree is used for the authentication purposes [3]. Pre-fetched based approach is used to identify and authenticate location of the consumer. The query validation and revaluation is done periodically on the server. The authentication problem is processed using subspace and arbitrary processing. It enables authentication for large dataset and for large subspaces. But this scheme failed to extent the work to networking model. The skyline process may not work in large network environment.

### 2.4 Authenticating k-nearest neighbor

This paper studies the problems in query verification on road networks [4]. This paper proposes network Voronoi verification scheme. The Voronoi cell is considered as the object to verify correctness and completeness of the queries. It reduces verification cost on mobile user by using Improved Distance verification model [23]. This verification model can support recent features in the mobile computing network. This paper can extend to work on more spacial regions. It is capable to support only one data-owner.

### 2.5 Cloud Centric Multi-level Authentication

It uses hierarchical authentication scheme of IOT devices on cloud network. It improves scalability on safety responsibility. It offloads continuous authentication and lightweight implementation [5]. It enables two level authentication on wireless sensor networks. This enables easier mobility management over the network. Elliptic Curve Cryptography is used for key encryption scheme [19]. Elliptic Curve Diffie Hellman algorithm is used for key exchange. Elliptic Curve Cryptography Digital Signature is used for digital signature generation. It is used for authenticating secret messages over the network.

### 2.6 Public Integrity Auditing for Cloud Sharing

It supports multi-user data sharing and modifications over the cloud. The protocol is very secured to implement and working [6][14]. But it fails to prove soundness and semantics of the implementing formula [26]. It revokes the client if not valid via third party auditing.

### 2.7 Anonymous and authentic data sharing system

It allows data-owner to authenticate anonymously and put into cloud storage for analysis. It is scalable for certificate verification for public key infrastructure [8]. It eliminates certificate based key verification using ID based ring structure. The secret key of the user and the digital signature that generated previously treated as an individual terms. This process eliminated data-owner re-authentication for every data. This scheme proves security pattern in low cost scheme.

### 2.8 Decentralized access control scheme

It supports anonymous authentication in cloud network. Without knowing the user identity the cloud authenticates the user data. It also provide access control which only valid user can decrypt and store data [9][15]. This scheme prevents the cloud from the reply attacks. This scheme is decentralized and robust. It supports communication, computation and proper storage.

### 2.9 Public Audibility and Data Dynamics

This paper achieves public auditing and Dynamic data processing [10]. It first verifies difficulties and potential security in doing this. Classic merkle hash tree construction is used for authenticating data on cloud [16]. Bilinear aggregate signature is used for multiple auditing. Third party auditing tool is used for multiple auditing simultaneously [17]. This system is highly secure and provable.

### 2.10 Object Centered Approach

It process the logging mechanism also with user policies and services [18]. Dynamic and traveling object is created for triggering access control and authentication that performed automatically on local [11]. This method proves efficiency and effectiveness of the system.

### 2.11 Role Based Cascaded Delegation

It supports simple and effective authority delegation over the servers on the cloud [12]. It enables user to create delegation rules based on his collaboration. This delegation role is verified by the administrative role [21]. The aggregated signature is used for authentication based on delegation rule.

### 2.12 k -times Attribute-Based Anonymous Access Control

Used to authenticate user anonymously on the cloud [13]. The cloud only knows about the attributes of the user not the ID of the user. These attributes are set with some limits and it is used to limit the users from some access. This scheme is proved as the practical one.

# 3. SYSTEM MODEL

## 3.1 Preliminaries

In this section the notations and parameters used in the paper is discussed.

*UI – User ID*

UI is provided for the client to authenticate himself as a valid user to the cloud. Every client is provided with the unique UI. UI in our proposed scheme is mention by 64 bits to 128 bits value which is registered by the user. The UI is mostly be the email id given by the user while on registration.

*UP – User Password*

UP is also an another parameter given by the client on registration maintained by the authentication server (AS) [14]. UP also vary from 64 bits to 128 bits based on client input.

*AS – Authentication Server*

AS is implemented in the cloud environment to authenticate the client in cloud and to provide him the user attribute. AS receives UI and UP from a client and authenticate the client.

*SP – Service provider*

SP is the one provides service for the client via the cloud. It verifies the client access to the services and then provides respective services to the client.

*UA – User Attribute*

UA is provided by the authentication server which is temporary for the client until the session expires. Unique UA is given to the client after the primary authentication. The UA is the 64 bits random string generated by the AS.

*AA – Access Attribute*

AA is provided by the SP to the client to access the web services. It is used to ensure that which service the client wanted to access.

*KA – Key Attribute*

KA is provided along with the AA to decrypt the file and access it. This attribute helps the client to access an service or file and decrypt it as per client use.

*SK – Session Key*

Session Key is to validate user access for particular service on the cloud. This attribute is provided for each and every service that can be accessed by the client.

*KF – Key Factor*

This is the encrypted key set provided to the client by the SP to validate the client and the access. The Key factor is the combination of UA, AA and KA.

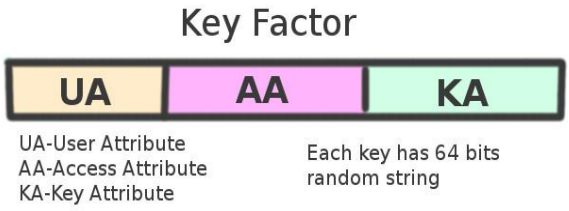


**Fig 1: Key Factor format**

Fig 1. shows the Key Factor used to send to the client. Total key size is 192 bits. First 64 bits of key represents User Attribute. Second 64 bits represents Access Attribute and Final 64 bits shows the Key Attribute.

The following table 1 shows the symbol and its definition

**Table 1: Symbol and definition**

| Symbol | Definition |
|---|---|
| UI | User ID |
| UP | User Password |
| AS | Authentication Server |
| SP | Service Provider |
| UA | User Attribute |
| AA | Access Attribute |
| KA | Key Attribute |
| SK | Session Key |
| KF | Key Factor |
| KFAAC | Key Factor Authentication and Access Control |
| HDD | Hard Disk Drive |
| IDE | Integrated Development Environment |

## 3.2 Proposed Model

The proposed model is name as the Key Factor Authentication and Access Control (KFAAC). With this proposed model the following features of the cloud can be improved or satisfied.

*Authenticity:* The Cloud should be properly authenticate the user to ensure whether the incoming client is valid user or not. The cloud server should not allow the invalid user inside the network. The authentication should be simple and not time consuming.

*Accessibility:* The Cloud should also capable of controlling access of users even though the client is the valid client. As the cloud is deserved as the pay per access or pay per service manner. The services are reserved for the particular clients only. A Client is limit to his access. This limitation should be verified every time of access.

Description:

The proposed model consisting of Client, Cloud Server, Service Provider Cloud and Authentication server. Initially a Client should register himself on the Cloud access his/her services from the cloud. On initial registration an access id and user id (UI) is provided to the client.

*Algorithm:*Client Authentication

*while true:*

*start server*

*get user_credentials*

*generate UI*

*store user_credentials,UI to DB*

*send UI to client*

*wait for new_client*

*resume*

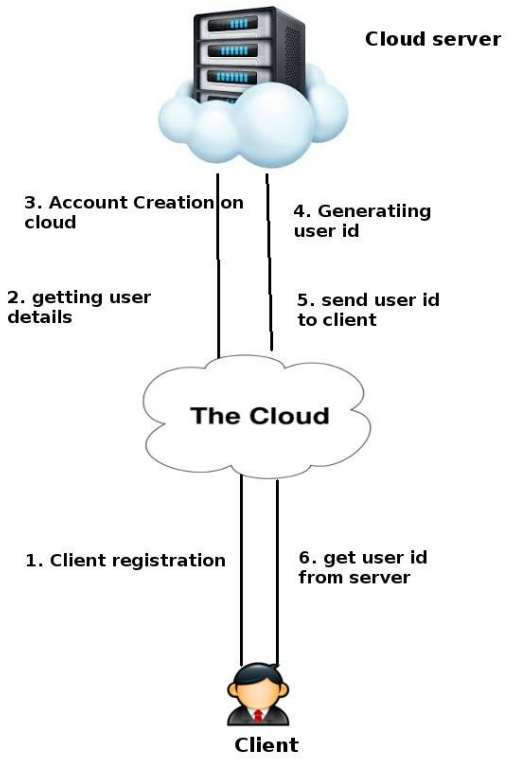


**Fig 2: Client Registration on Cloud**

*Algorithm: Client Authentication*
*while true:*
*wait for authentication*
*get UI from client*
*get password from client*
*get db_password for UI*
*if db_password = password:*
*generate session_key*
*send session_key*
*else*
*invalidate_client*

The next step of the client is to authenticate himself to enter into the cloud environment. To do so the client has to connected to the Authentication server from the cloud network. Then the client has to submit his/her user id and the password to the authentication server. The authentication server verifies the user id and the password and generates temporary session key for that particular client. Then the client receives the session key from the authentication server to continues his/her session. The session key will not be provided for the invalid authentication of client.

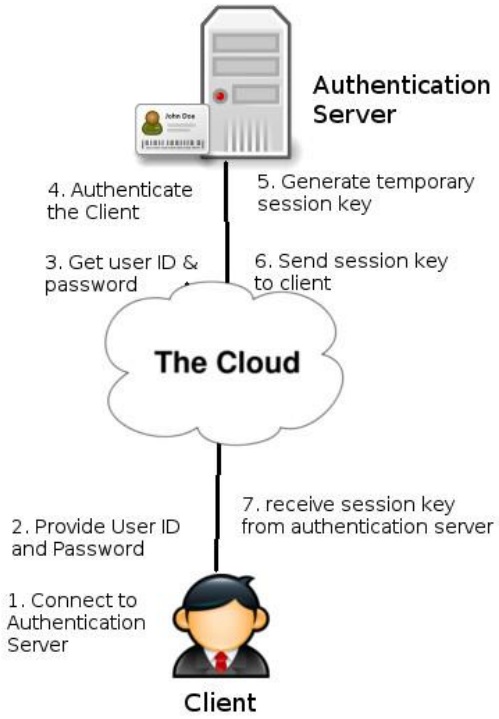


**Fig 3: Client Authentication on Server**

In addition to the authentication process the Authentication server also verifies the client additional details like location, IP address, Hardware Address, Browser name, Operating System time and Time of usage. These credentials are used for the future verification of the client to ensure him/herself as the valid client.

*Algorithm: Client Access Validation*
*while true:*
*get UI and session_key*
*get_all UI list*
*for id in UI_list*
*if id=UI*
*generate access_key*
*send access_key*
*else*
*ommit client*
*end for*

The next of the process is to validate access of the client. The client must connect to the access server next. Then the client should bind its user id and session key and sent it to the Access Server. The Access Server validates the client's user id and the session id and then generates the access id for the valid client. And sends that access id to the client. Using this access id the client can request for services from the service provider.

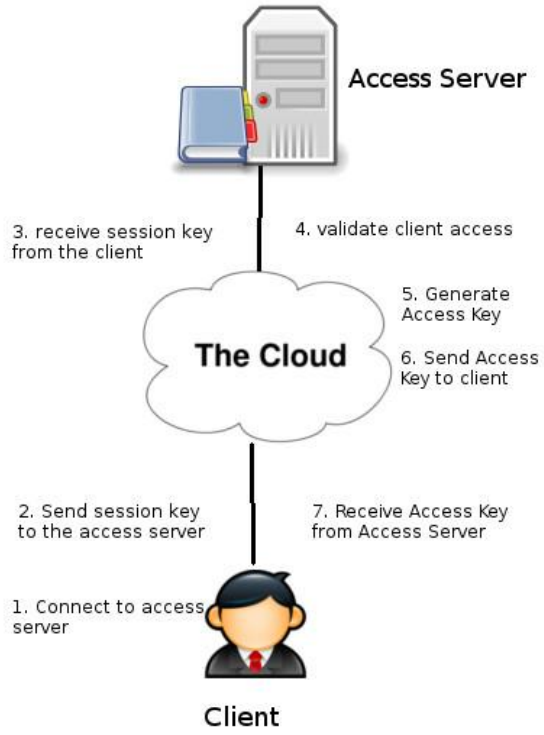


**Fig 4: Client Access Verification on Cloud**

*Algorithm: Key Factor Authentication thread_1*

*while true:*

*get request*

*parse request*

*parameters[]← extract_parameter(request)*

*for value in parameters:*

*get access_key*

*end for*

*if access_key is valid*

*generate access_id*

*generate key_factor*

*key_factor← user_id+access_id+key_id*

*encrypt key_factor with access_key*

*send key_factor*

*else*

*reject access*

*Algorithm: Key Factor Authentication thread_2*

*while true:*

*get access_request*

*parse access_request*

*extract access_id*

*validate access_id*

*if access_id=server_access_id*

*provide access_id*

*else*

*discard_user_id*

From the previous step the client will be having the access key to access the service provider. The service provider has to verify the access of the client to the server. The client must send his/her access key to the service provider. Using this access key the service provider generate and encrypt the key factor for the client. Then encrypts the key factor using the access key of the client. Then the encrypted key factor is sent to the client. Only the valid client can decrypt the key factor and extract the access id from the key factor. Then using this access id the client can request for the web service. The service provider validates the access id, user id and session id of the client to provide the service from the cloud.

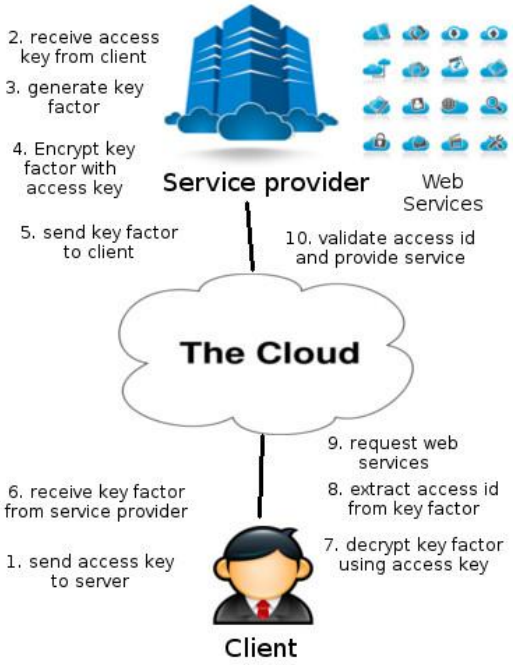


**Fig 5: key factor authentication for client**

# 4. EXPERIMENTAL EVALUATION

## 4.1 Experimental Setup

The experimental environment is set under Windows 7 Ultimate Operating System. JDK 1.8 is used as the development platform. We use Netbeans 8.1 IDE to develop experimental applications. We developed three web application which can communicate with each other using common database. These servers are developed using Servlet and JSP code and implemented using Tomcat server. The server are authentication server, Access Server and Service provider. All three servers are implemented in deferent PC of following configurations. The server PC has Intel Dual Core processor with 4 GB of RAM memory with 250 GB internal HDD. The PCs are connected as a local LAN network with the backbone of ADSL Router. The network is with the speed of 100MBPS.

Experimental Result

All three servers are deployed one by one on the LAN network to form a sample cloud environment. For testing

purpose 200 client accounts is created for the cloud server. Every client account has its desired unique id and access rights. Clients are asked to access the web-services from the cloud. The authentication server is capable to authenticate all the clients at a time with this limited RAM and network. The user authentication is done by server accurately for every authentication request. The Access server performed well in generating access key for authenticated clients. The client access control is perfected with the user of key factor authentication. So there can't be any unauthenticated access on the cloud is 0% possible only.

This scheme is suitable for distributed cloud computing environment. It supports anonymity and user privacy. Existing systems are vulnerable to reply attacks but in proposed scheme reply attacks is not possible to perform. The proposed scheme is also prone to synchronization problems. Our proposed algorithm is also secured against the forgery problems [20]. An attacker may use password guessing attacks but cannot access the services without the valid access key.

**Table 2 shows the Comparison of two and key factors**

| Feautures | Two Factor Authentication | Key Factor Authentication |
|---|---|---|
| Resistance to reply attack | Yes | Yes |
| Preserve User Anonymity | No | Yes |
| User Traceability | Yes | Yes |
| Preventing Password Attacks | No | Yes |
| Synchronization problem prevention | No | Yes |
| Forgery Attack prevention | Yes | Yes |
| Suitable for All Service Providers | No | Yes |
| Able to control mutisession | No | Yes |
| Preserve Access Security | No | Yes |
| Service Continuety | Yes | Yes |
| Servce Switching | No | Yes |

# 5. CONCLUSION

The proposed scheme is used to authenticate and access the client on the distributed cloud computing environment. This scheme allows the client to access services from various service provider on the cloud in secured way. It provides privacy for both the service provider and the client and also the cloud environment. The proposed scheme authenticates the client using the Key Factor authentication scheme. It validates the user session, access key and the provides the service only if the client can able to decode the access id provided by the access server. The scheme provides user anonymity, service security, secured accessibility and flawless services. Security analysis shows that the proposed scheme can withstand major security challenges.

# 6. FUTURE ENHANCEMENT

Implementation can to modified for maintain individuality of the client on the network. The authentication process can be skipped by ensuring the security. But the temporary client who has no account can also consume the pay per use service in s secured accessible way. The security can be improved on DOS attacks which may cause the authentication process slower on the network. Algorithm on key generation can be improved for the future needs. Implementing Triple DES may improve the security challenges of providing services to the malicious valid client.

# 7. REFERENCES

[1] Chunsheng Zhu, Student Member, IEEE, Hasen Nicanfar, Student Member, IEEE, Victor C. M. Leung, Fellow, IEEE, and Laurence T. Yang, Member, IEEE, "An Authenticated Trust and Reputation Calculation and Management System for Cloud and Sensor Networks Integration", IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, OCTOBER 2014.

[2] Jia-Lun Tsai and Nai-Wei Lo, " A Privacy-Aware Authentication Scheme for Distributed Mobile Cloud Computing Services", IEEE SYSTEMS JOURNAL, 2015.

[3] Xin Lin, Jianliang Xu, Senior Member, IEEE, Haibo Hu, and Wang-Chien Lee, "Authenticating Location-Based Skyline Queries in Arbitrary Subspaces", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 6, JUNE 2014.

[4] Yinan Jing, Member, IEEE, Ling Hu, Wei-Shinn Ku, Senior Member, IEEE, and Cyrus Shahabi, "Authentication of k Nearest Neighbor Query on Road Networks", Yinan Jing, Member, IEEE, Ling Hu, Wei-Shinn Ku, Senior Member, IEEE, and Cyrus Shahabi.

[5] Ismail Butun, Melike Erol-Kantarci, Burak Kantarci, and Houbing Song, "Cloud-Centric Multi-Level Authentication as a Service for Secure Public Safety Device Networks", Public Safety Networks, 2016.

[6] Yong Yu, Yannan Li, Jianbing Ni, Guomin Yang, Yi Mu and Willy Susilo, "Comments on Public Integrity Auditing for Dynamic Data Sharing with Multi-user Modification", IEEE Transactions on Information Forensics and Security, 2015.

[7] Xinyi Huang, Joseph K. Liu , Shaohua Tang, Yang Xiang, Kaitai Liang, Li Xu, Jianying Zhou, "Cost-Effective Authentic and Anonymous Data Sharing with Forward Security", IEEE Transactions on Computers, 2013.

[8] Sushmita Ruj, Member, IEEE, Milos Stojmenovic, Member, IEEE, and Amiya Nayak, Senior Member, IEEE, "Decentralized Access Control with Anonymous Authentication of Data Stored in Clouds", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 25, NO. 2, FEBRUARY 2014.

[9] Qian Wang, Student Member, IEEE, Cong Wang, Student Member, IEEE, Kui Ren, Member, IEEE,Wenjing Lou, Senior Member, IEEE, and Jin Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", IEEE TRANSACTIONS ON

PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 22, NO. 5, MAY 2011.

[10] Smitha Sundareswaran, Anna C. Squicciarini, Member, IEEE, and Dan Lin, "Ensuring Distributed Accountability for Data Sharing in the Cloud", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 9, NO. 4, JULY/AUGUST 2012.

[11] Roberto Tamassia, Fellow, IEEE, Danfeng Yao, Member, IEEE, and William H. Winsborough, "Independently Verifiable Decentralized Role-Based Delegation", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS, VOL. 40, NO. 6, NOVEMBER 2010.

[12] Tsz Hon Yuen, Joseph K. Liu , Man Ho Au, Xinyi Huang, Willy Susilo, Jianying Zhou, "k -times Attribute-Based Anonymous Access Control for Cloud Computing", IEEE Transactions on Computers, 2013.

[13] Shuanghe Peng, Zhige Chen, Deen Chen, "Membership Proof and Verification in Authenticated Skip Lists Based on Heap", S ECURITY S CHEMES AND S OLUTIONS, 2016.

[14] Chang Liu, Rajiv Ranjan, Chi Yang, Xuyun Zhang, Lizhe Wang, Senior Member, IEEE, and Jinjun Chen, Senior Member, IEEE, "MuR-DPA: Top-down Levelled Multi-replica Merkle Hash Tree Based Secure Public Auditing for Dynamic Big Data Storage on Cloud", IEEE TRANSACTIONS ON COMPUTERS, 2013.

[15] Igor Faynberg, Hui-Lan Lu, and Herbert Ristock, "On Dynamic Access Control in Web 2.0 and Beyond: Trends and Technologies", 2011.

[16] Jin Li, Xiao Tan, Xiaofeng Chen, Duncan S. Wong, Fatos Xhafa, "OPoR: Enabling Proof of Retrievability in Cloud Computing with Resource-Constrained Devices", IEEE Transactions on Cloud Computing, 2013.

[17] Jiawei Yuan, Shucheng Yu, Member, IEEE, "Public Integrity Auditing for Dynamic Data Sharing with Multi-User Modification", IEEE Transactions on Information Forensics and Security, 2013.

[18] Danan Thilakanathan, Shiping Chen, Surya Nepal and Rafael Calvo, "SafeProtect: Controlled Data Sharing with User-Defined Policies in Cloud-based Collaborative Environment", JOURNAL OF L A TEX CLASS FILES, VOL. 11, NO. 4, DECEMBER 2012.

[19] Luca Ferretti, Fabio Pierazzi, Michele Colajanni, and Mirco Marchetti, "Scalable architecture for multi-user encrypted SQL operations on cloud database services", IEEE Transactions on Cloud Computing, 2013.

[20] Fei Chen, Tao Xiang, Yuanyuan Yang, and Sherman S. M. Chow, "Secure Cloud Storage Meets with Secure Network Coding", IEEE Transactions on Computers, 2015.

[21] Indrajit Ray, Kirill Belyaev, Mikhail Strizhov, Dieudonne Mulamba, and Mariappan Rajaram, "Secure Logging As a Service—Delegating Log Management to the Cloud", IEEE SYSTEMS JOURNAL, 2011.

[22] Gongjun Yan, Ding Wen, Stephan Olariu, and Michele C. Weigle, "Security Challenges in Vehicular Cloud Computing", IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 14, NO. 1, MARCH 2013.

[23] Honggang Wang, University of Massachusetts Shaoen Wu, Ball State University Min Chen, Huazhong University of Science and Technology Wei Wang, South Dakota State University, "Security Protection between Users and the Mobile Media Cloud", IEEE Communications Magazine, March 2014.

[24] Hong Liu, Student Member, IEEE, Huansheng Ning, Senior Member, IEEE, Qingxu Xiong, Member, IEEE, and Laurence T. Yang, Member, IEEE, "Shared Authority Based Privacy-preserving Authentication Protocol in Cloud Computing", IEEE Transactions on Parallel and Distributed Systems, 2013.

[25] Slawomir Grzonkowski and Peter M. Corcoran, Fellow, IEEE, "Sharing Cloud Services: User Authentication for Social Enhancement of Home Networking", IEEE Transactions on Consumer Electronics, Vol. 57, No. 3, August 2011.

[26] T.Thamarai selvan,Glidersoft, "Nadi Aridhal: A pulse based automated diagnostic system",Electronics Computer Technology (ICECT), 2011 3rd International Conference,April 2011.

[27] Ling Hu, Student Member, IEEE, Wei-Shinn Ku, Senior Member, IEEE, Spiridon Bakiras, and Cyrus Shahabi, Senior Member, IEEE, "Spatial Query Integrity with Voronoi Neighbors", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 4, APRIL 2013.

[28] IBRAHIM LAHMER AND NING ZHANG, "Towards a Virtual Domain Based Authentication on MapReduce", 2016.