Improved Security of Audit Trail Logs in Multi-Tenant Cloud Using ABE Schemes

Bhanu Prakash Gopularam
Cisco Systems India Pvt. Ltd
Department of Computer Science and Engineering
Nitte Meenakshi Institute of Technology
Bangalore, India

Nalini N
Department of Computer Science and Engineering
Nitte Meenakshi Institute of Technology
Bangalore, India

***Abstract*—Cloud computing is delivery of services rather than a product and among different cloud deployment models, the public cloud provides improved scalability and cost reduction when compared to others. Security and privacy of data is one of the key factors in transitioning to cloud. Typically the cloud providers have a demilitarized zone protecting the data center along with a reverse proxy setup. The reverse proxy gateway acts as initial access point and provides additional capabilities like load balancing, caching, security monitoring capturing events, syslogs related to hosts residing in the cloud. The audit-trail logs captured by reverse proxy server comprise important information related to all the tenants. While the PKI infrastructure works in cloud scenario it becomes cumbersome from manageability point of view and they lack flexibility in providing controlled access to data. In this paper we evaluate risks associated with security and privacy of audit logs produced by reverse proxy server. We provide a two-phase approach for sharing the audit-logs with users allowing fine-grained access. In this paper we evaluate certain Identity-Based and Attribute-Based Encryption schemes and provide detailed analysis on performance.**

***Keywords*—multi-tenancy; audit-trail log; attribute-based encryption; reverse proxy security**

I. Introduction

Cloud computing as defined by NIST is a model for enabling convenient, on-demand network access to a shared pool of configurable resources that can be rapidly provisioned and releases with minimal management effort or interaction. While the private cloud gives organizations greater control over the infrastructure it may not be cost effective for small and medium businesses [1]. Cloud services are offered in different service models and three well known models are Infrastructure as a Service – Cloud user has greater control on infrastructure Vmware, Openstack, Azure offer such services. Platform as a Service – developer centric services Heroku, Google AppEngine are few providers, and Software as a Service – services include data analytics, online meetings such as Cisco WebEx, Gmail. Cost reduction and increased efficiency are primary motivations towards a public cloud and nevertheless security and privacy objectives play vital role for decisions about outsourcing IT services [2]. The data collected by network devices such as firewalls, reverse proxy servers, hypervisor are very vital for monitoring health of cloud as well as for security forensics [3].

The internet facing reverse proxy gateway provides protection from issues like intrusion detection, denial of service attacks etc. Data collected by reverse proxy includes system logs, alarms and it can capture HTTP/REST requests, remote-service calls pertaining to tenants if it is configured as SSL termination end-point. The traditional way of log sharing suffers from few problems like:

1) Existing PKI based techniques for preserving the audit logs largely relay on certificates for exchanging the data. Considering cloud storage as untrusted, managing centralized repository for certificates of multitude of tenants necessities frequent synchronization with key servers and the process is error-prone due to large number of interactions with PKG server.

2) The traditional PKI infrastructure either reveals all the data or restricts and does not provide easy way to allow fine-grained access to data considering organizational policy information.

II. Key Contributions

In this paper we outline challenges associated with audit-log preservation in cloud with reverse proxy architecture. We experiment with advances in attribute-based encryption schemes to overcome privacy and security problem of audit trail logs. We experiment with Identity-based encryption techniques proposed in [7] referred as *BB* scheme here by and *committed blind anonymous* identity-based encryption as proposed in [8] referred as *CKRS* scheme here by.

Ciphertext Policy Attribute Based Encryption techniques proposed in [9] referred as *BSW* scheme here by and *efficient and secure realization* of CP-ABE scheme proposed in [10] referred as *Waters* scheme here by.

In literature the above schemes are also considered as key milestones in the identity-based cryptography. We provide a work flow for secure distribution of audit-trail logs captured by reverse proxy server among multiple tenants. We evaluate the performance of operations like setup, key-generation, encryption, decryption under various configurations. Few applications of proposed scheme include secure reverse proxy implementation without overhead of certificate management, enabling on-demand third party auditing or inspection, and secure sharing of logs with interested parties with fine-grained access control.

III. Preliminaries

A. Identity and Attribute Based encryption:

Shamir first proposed concept of Identity-based public key cryptography in mid 1980 and in 2001 the first practical and secure IBE scheme [12] was presented by Boneh and Franklin. Sahai and Waters [14] first introduced concept of Attribute-Based encryption in 2006, the user attributes were used to encrypt and decrypt data. In the same year Bethencourt et al. [9] presented first construction of Ciphertext-Policy Attribute-Based encryption. Using CP-ABE it is possible to embed role based access control policies into the ciphertext. Later attribute based encryption was extended to distributed identities and hierarchical attribute based encryption schemes. ABE systems now support many crucial functionality [15] required by security infrastructure.

B. Reverse Proxy Gateway

A reverse proxy is server side software typically acts as entry point for HTTP requests. Typically reverse proxy resides in DMZ facing the internet. The HTTP request is scrutinized first and requested content is served if it is already in cache or statically referred. This setup is compelling for cloud service providers with multi-tenancy architecture and having a single entry point with capability to route requests provide lots of benefits for CSPs. Other important usecases include B2B transactions, supply chain integration. Some of the key functionality provided by reverse proxy gateway architecture is described here. One can refer reverse proxy websites like Nginx [4], SkyHigh software architecture to know more.

a) Security: Reverse proxy can provide single point of communication. It can decrypt the HTTPS based request and communicate with back end servers in HTTP mode. Provides many advantages for cloud users like ease of configuration of SSL/TLS, saves CPU intensive security operations using specialized hardware.

b) Centralized Logging and Auditing: As all HTTP requests are routed through reverse proxy server, it captures all the important events related to hosts residing in the cloud.

c) Load balancing: RP can route the incoming HTTP requests among the available servers using strategies like round robin, sticky session in case of stateful sessions etc.

d) Caching and serving static content: For storage based cloud applications viz., youtube, vimeo the server responsiveness can be improved by hosting static content and using RP for routing.

C. Audit Trail Log Structure

Reverse proxy can be configured to generate logs like file, stderr or syslogs. For example in Nginx server, the log format is specified using log_format directive

```
http {
 log_format compression
'$remote_addr - $remote_user [$time_local] '
'"$request" $status $body_bytes_sent '
'"$http_referer" "$http_user_agent";
}
```

TABLE I. AUDIT-TRAIL LOGS GENERATED BY REVERSE PROXY GATEWAY

Attribute Name	User Login Activity	Resource Access Activity
Time	14:14:19.566	12:13:26.080
UserID	Supervisor801	admin
EventType	User Logging	User Access
EventStatus	Failure	Success
ClientAddress	https:64.103.237.53 :tcp:54665	64.103.237.53
ResourceAccessed	AppAdmin	Channel Provider/2
CompulsoryEvent	Yes	No
ComponentID	Administration	Configuration API
AuditCategory	Authentication attempt failed	channelProvider/2 2 modified
AppId	10023	1055
ClusterId	1	1
NodeId	uccx-93-55	uccx-93-55

IV. CHALLENGES IN PRESERVING AUDIT-TRAIL LOGS IN MULTI-TENANCY CLOUD

Besides many potential benefits the public cloud the data security is complex due to following challenges

- *Shared Multi-tenant Environment* – Public cloud achieves multi-tenancy by logical separation at multiple layers of software stack. The attacker can pose as a consumer and exploit vulnerabilities from cloud environment.
- *Loss of Control* – While the cloud users may perceive the services as traditional service model, transitioning of control to cloud provider amplifies the risks associated.
- *System complexity* – Complexity largely depends on infrastructure used and often the cloud providers use methods that are proprietary in nature. Typically complexity relates inversely to security, the complexity leads to increased risk for vulnerabilities.
- *Audit trail logs* – the cloud computing environment poses new challenges from audit and monitoring perspective. Full audit trail within the cloud is still an open problem and poses lots of challenges as seldom organization security policy challenges does meet the cloud provider practices.

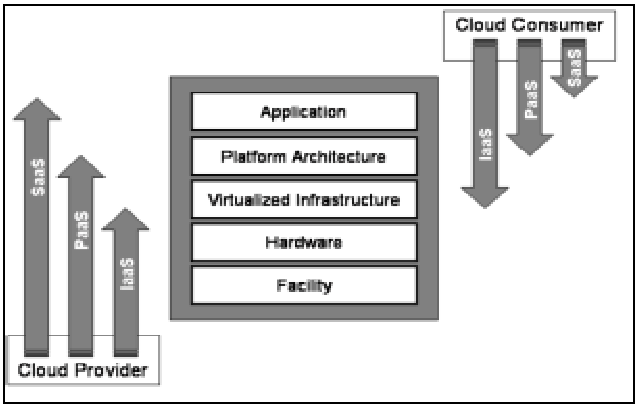

Fig. 1. Cloud service models and differences in scope and control

V. Methodology

Consider public cloud provider having multiple tenants and protected with reverse proxy server which captures audit-trail records of incoming traffic. We consider role of reverse proxy server extended as SSL termination end-point so that it can intercept all HTTP/SSL traffic. The cloud provider has a Network Admin who has access to entire logs and cloud tenants with users having roles like level-1, level-2, level-3 etc. While level-1 users are in the bottom of organizational hierarchy and they are monitored by level-2 and so on and so forth.

A. Privacy and Security of Audit logs - Objectives

We divide the problem into two sub-domains – 1. Cloud Network Admin has access control on entire logs and can do operations like key-generation, encryption, decryption, 2. Tenant users like Network Admin can access all tenant specific logs and users of Level-1, Level-2 etc. has controlled access to data. Users at higher level can oversee data pertaining to lower level that they are administering. It implies that user's access to audit log contents is controlled using *role-based access control* policies.

TABLE II. CLOUD USERS ACCESS TO CONTENTS OF AUDIT LOGS CATEGORIZED INTO TYPE-1, TYPE-2 SECURITY

Participant Role	Accessible content in audit-trail log	Category
Level-1 [Tenant]	Time, UserID, EventType	Type-1
Level-2[Tenant]	Time, UserID, EventType, EventStatus, ClientAddress, ResourceAccessed	
Network Admin [Tenant]	Time, UserID, EventType, EventStatus, ClientAddress, ResourceAccessed, CompulsoryEvent, ComponentID	
Cloud Network Admin [Cloud Provider]	Time, UserID, EventType, EventStatus, ClientAddress, ResourceAccessed, CompulsoryEvent, ComponentID, AuditCategory, AppId, ClusterId, NodeId	Type-2

B. Design

For audit-trail log security we choose 2-phase protection. The unique challenge here is that the security mechanisms should ensure that cloud providers has complete control on the data and has ability to share with tenants and while restricting access according to organizational hierarchy. We solve this problem using blend of identity and attribute-based encryption schemes. The problem is solved in 2-phase approach

a) Type-I Data Security

Type-I data protection involves security mechanism like Identity-based encryption [7] [8]. The Cloud Network Admin has access to all the data but individual tenants should have access to their data only. We use identity-based encryption scheme for access control. Each encrypted log entry is associated with public identifiers or tags like *TenantId* and user keys are associated with access policy. Although entire logs are kept in shared location in cloud, the individual tenants can access only their data. The reason for choosing identity-based encryption scheme is that it is possible to share data without requiring exchange of certificates. We evaluated two identity-based encryption schemes [7] [8] for performance with large datasets.

b) Type-II Data Security

Type-II data security involves allowing fine-grained access control on data to tenant users. The user can decrypt data only if the attributes in secret key satisfies the access structure of encrypted data. For example Level-1 user can see only her own activity while the Level-2 can see activity of all his employees and soon. We use ciphertext-policy attribute based encryption schemes [9][10] with ciphertext having policy information of participants and user keys having descriptive attributes about participant. The reason for choosing CP-ABE scheme here is that it is perfectly suited for environment where user privileges (*role-based access control*) determine the access to data and it allows fine-grained access control on the data. We experiment with *BSW* [9] and *Waters* scheme [10] for performance.

Depending on the log sharing mechanism two possible approaches exist.

- Cloud provider use Type-1 security mechanism for logs encryption and Cloud tenants access their data and decrypt and re-encrypt using Type-2 security mechanism
- Cloud provider applies Type-2 security mechanism which internally uses policy tree for log encryption and then re-encrypt using Type-1 security mechanism. The tenants access the data by using Type-1 secret keys and then use Type-2 secret keys to decode the data.

In this paper we provide experimental results of Type-1 and Type-2 security mechanisms separately and results are applicable in both the cases outlined.

c) Setup and Key Generation

- Type-1: The algorithm initialization depends on bilinear pairing and elliptic curve used. The master secret *MK* and public key *PK* are generated using system parameters *P*.

- Type-2: This can be done by cloud provider or tenant itself depending on use case. The CP-ABE *Setup(k)* is run with security parameter and it results in public parameters (*PK*) and master key (*MK*). The CP-ABE KeyGen(*MK*, *PK*, T) with possible tenant-id values which outputs decryption keys associated with attributes.

d) Encryption and Decryption:

- Type-1: Each log entry, the data <ApplicationId, ClusterId, NodeId> is encrypted using a symmetric key algorithm and using individual tenant-id t_i as public key the server computes ciphertext c_i of the data and P as public parameters. Here P may be equal to t_i if cloud provider wishes to annotate with tenant-id only. Data is decrypted using (*PK*, *sk*, CT)

- Type-2: For each log record, the data pertaining to tenants, the Enc(*Record*, T, *PK*) where T relates to access structure T for public parameters *PK*. The CP-ABE Dec(CT, *SK*, *PK*) is run using user secret keys *SK* and public parameters *PK*.

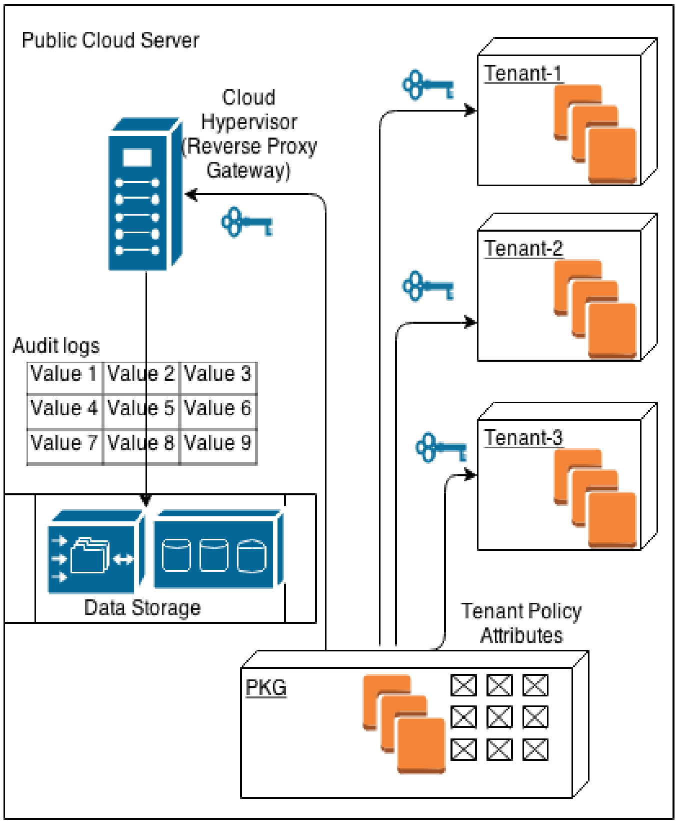

Fig. 2. Multi-Tenant cloud with audit trail mechanism secured using combination of ABE

VI. Experiments and Evaluation

We use a hypothetical example of public cloud provider hosting 3 tenants.

A. *System Details-We have used CHARM crypto-library[5] [6] v0.43 for prototyping. At a very high-level the library provides a protocol engine for many cryptographic operations and an adapter architecture which bridges gaps necessary for building a complete crypto system. In addition we used other open source libraries including OpenSSL 1.0.1, GMP 6.0.0a and Pairing-Based Cryptography library version 0.5.14 of Stanford. The experiments were carried on X86 based platform using Ubuntu 12.04.4 LTS (precise) 32-bit server with 8 GB RAM and Intel Core i5-3470 CPU with 3.2 GHz 4 core processor.*

B. *Test Data - The sample audit-trail logs used in experiments is sampled from a reverse proxy server. The dataset is split into chunks of approximately 20000 records carefully having activity of cloud tenants with possible operations. We analyze performance of cryptographic schemes with these chunks.*

C. *Data Security – Results - We have used elliptic curve with bilinear maps (or pairings) like 512 bit symmetric curve. We used Type-A curve such as* $y^2 = x^3 + x$ *to compute the pairings. The secret key is communicated to interested parties using a secure channel like TLS/SSL*

TABLE III. Setup Time for Type-1 Security (IBE schemes)

Operation	Scheme	Time (milliseconds)
Setup	Ibe-bb[a]	15.624
	Ibe-ckrs[a]	52.361

a. For pairings symmetric curve with 512 bit is used

TABLE IV. Key Generation Time for Type-1 Security

Operation	Scheme	Time (milliseconds)
Key Generation	Ibe-bb	3.137
	Ibe-ckrs	22.689

While implementing Type-1 security, the *CKRS* scheme took time prohibitively large time than *BB* scheme for initial setup (master key and public key generation) and secret key generation.

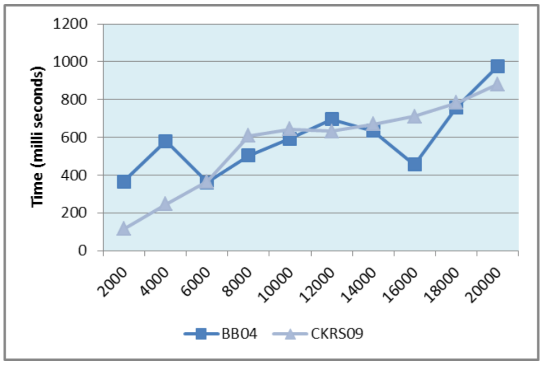

Fig. 3. Encryption using Type-1 (IBE schemes)

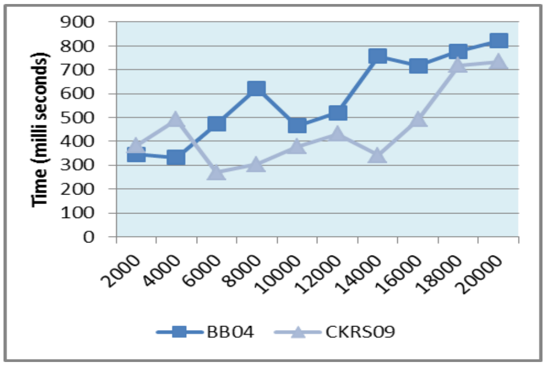

Fig. 4. Decryption using Type-1 (IBE schemes)

The encryption and decryption of data using *CKRS* scheme was more performant (5-10%) then *BB* scheme with large datasets. The encryption involves generating a random symmetric key using pairing and encrypting the data using symmetric crypto system such as AES in CBC mode with 16 byte block size. Then the symmetric key is encrypted using IBE algorithm.

D. Type-2 Data Security – Results

The initial setup time for *BSW* scheme was approximately twice the *WATERS* scheme initialization. And secret key generation time for level-1 and level-2 users of cloud tenant with *BSW* and *WATERS* scheme was roughly same.

TABLE V. SETUP TIME FOR TYPE-2 SECURITY (CPABE SCHEMES)

Operation	**Scheme**	**Time (milliseconds)**
Setup	cpabe-bsw[b]	38.305
	cpabe-waters[b]	21.2

b. For pairings symmetric curve with 512 bit is used

TABLE VI. KEY GENERATION TIME FOR TYPE-2 SECURITY

Operation	**Scheme**	**Level-2 key**	**Level-1 key**
Key generation	cpabe-bsw	23.339	23.467
	cpabe-waters	24.569	24.404

The *BSW* scheme took comparatively more time for encryption and decryption with large datasets than *WATERS* scheme and *WATERS* scheme performance was quite stable.

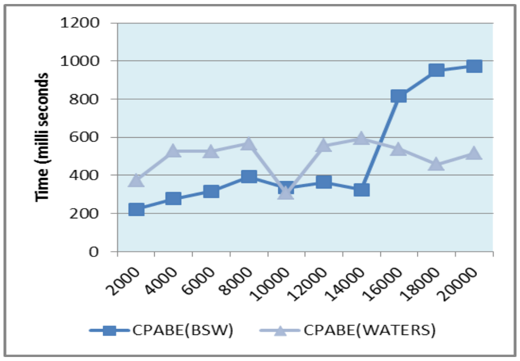

Fig. 5. Encryption using Type-2

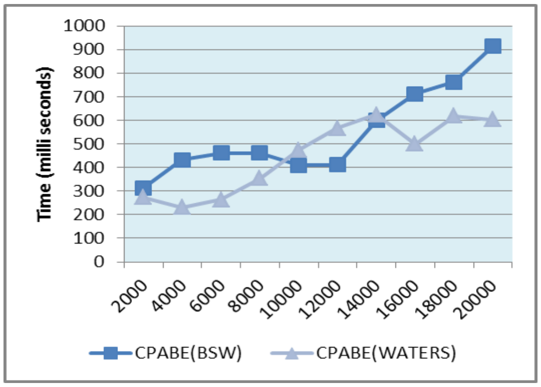

Fig. 6. Decryption by Level-2 tenant user using Type-2

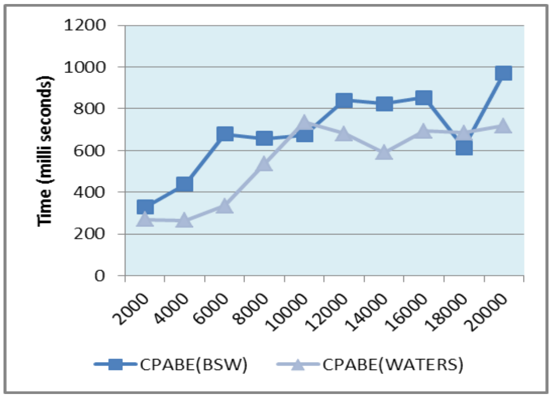

Fig. 7. Decryption by Level-1 tenant user using Type-2

The proposed scheme provides security of sensitive data and provides fine-grained access control to the data and following are some limitations of identity based systems.

1) A unique characteristic of identity based systems that differentiates from existing PKI schemes is that the encryption is possible without any need for communicating with server during validity period of the public parameters. This reduces network communication significantly but can lead to problems in case of lost key or key revocation. Given the fact that identity based systems require lesser validation with key servers, the IBE private keys should not be created using timestamp of longer duration as this could worsen the problem of key compromise.

2) Using identity system based on bilinear pairings devised on family of supersingular elliptic curves over finite fields (also called as type-1 curves) for practical applications using Advanced Encryption Standard keys it is sufficient to use 128-bit levels and higher such as 192 bits or 256 bits.

VII. CONCLUSION AND FUTURE WORKS

Audit log preservation in cloud is a challenging problem considering the dynamicity of cloud. The current mechanisms to share the logs securely involve large overhead in terms of certificate management and do not offer flexibility to share data. The combination of different identity-based encryption techniques discussed in this paper provide a simpler mechanism for log-sharing to intended receivers. In future we plan to extend the research to implementing oblivious search on encrypted audit logs along with computation on data like analytics with monotonic and non-monotonic access structures and along with predicate encryption.

ACKNOWLEDGMENT

We would like to thank Sashank Dara and *crypto.stackexchange* community for insightful discussions and suggestions. We would also like to thank Satheesh Kumar and Sridhar Gaddipati, CBABU, Cisco Systems for supporting us while carrying this work.

REFERENCES

[1] Michael Armbrust et al, "Above the Clouds: A Berkeley View of Cloud Computing", Technical Report No. UCB/EECS-2009-28, February 10, 2009

[2] Siani Pearson and Azzedine Benameur, "Privacy, Security and Trust Issues Arising from Cloud Computing", 2nd IEEE International Conference on Cloud Computing Technology and Science, HP Labs, pp. 693-702

[3] Binti Abdul Aziz, N, Binti Meor Yusoff, N.D, Binti Abu Talib, "Log Visualization of Intrusion and Prevention Reverse Proxy Server against Web Attacks", Informatics and Creative Multimedia (ICICM), International Conference, 2013, pp. 325-329

[4] Wei Yuan, Hailong Sun, Xu Wang, Xudong Liu, "Towards Efficient Deployment of Cloud Applications through Dynamic Reverse Proxy Optimization", High Performance Computing and Communications, IEEE, 2013, pp. 651 - 658

[5] Yannis Rouselakis, Brent Waters, "Practical constructions and new proof methods for large universe attribute-based encryption", ACM SIGSAC conference on Computer & communications security, 2013, pp. 463-474

[6] Joseph A Akinyele, Christina Garman, Ian Miers, Matthew W Pagano, Michael Rushanan, Matthew Green, Aviel D Rubin, "Charm: A framework for rapidly prototyping cryptosystems", Journal of Cryptographic Engineering, Springer-Verlag, 2013, pp. 111-128

[7] Dan Boneh , Xavier Boyen, "Efficient Selective-ID Secure Identity Based Encryption Without Random Oracles", Proceedings of Eurocrypt 2004, volume 3027 of LNCS, 2004, pp. 223-238

[8] Jan Camenisch , Markulf Kohlweiss , Alfredo Rial , Caroline Sheedy, "Blind and Anonymous Identity-Based Encryption and Authorised Private Searches on Public Key Encrypted Data", PKC 2009

[9] John Bethencourt, Amit Sahai, and Brent Waters, "Ciphertext-policy attribute-based encryption", 28th IEEE Symposium on Security and Privacy, Oakland, May 2006 , pp. 321-334

[10] Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in Public Key Cryptography - PKC 2011 , Vol. 6572, 2011, pp. 53–70.

[11] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log", 11th Annual Network and Distributed System Security Symposium 2004

[12] Boneh, D., Franklin, M, "Identity-Based Encryption from the Weil Pairing", Kilian, J. (ed.) CRYPTO 2001. Springer LNCS, vol. 2139, pp. 213–229

[13] V. Goyal, O. Pandey, A. Sahai and B. Waters, "Attribute Based Encryption for Fine-Grained Access Conrol of Encrypted Data", ACM conference on Computer and Communications Security, 2006

[14] Sahai and B. Waters, "Fuzzy Identity Based Encryption", IACR ePrint Archive, Report 2004/086

[15] Schridde, C, Dornemann, T, Juhnke, E, Freisleben, Smith, M."An identity-based security infrastructure for Cloud environments", IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS), 2010 pp. 644-649