A Survey of Various Techniques to Secure Cloud Storage

Satyendra Singh Rawat[†], **Prof. Niresh Sharma**[††],

[†]M.Tech Research Scholar RKDF Institute of Sc. & Technology Bhopal, India
[††]HOD Computer Sc. & Engg.. RKDF Institute of Sc. & Technology Bhopal, India

Abstract
When it comes to cloud data protection methods, no particularly new technique is required. Protecting data in the cloud can be similar to protecting data within a traditional data center. Authentication and identity, access control, encryption, secure deletion, integrity checking, and data masking are all data protection methods that have applicability in cloud computing. This paper will briefly review few methods and will note anything that is particularly unique to when these are deployed in a cloud.

Keywords
Cloud Computing, Cloud Security, Encryption, Role-based Access Control.

1. Introduction

In cloud storage systems , the server that stores the client's data is not necessarily trusted. Therefore, users would like to check if their data has been tampered with or deleted. However , outsourcing the storage of very large files(or whole file systems) to remote servers presents an additional contrast: the client should not download all stored data in order to validate it since this may be prohibitive in terms od bandwidth and time.

A Proofs of retrievability (PORs) [2][5][4][6][1][3][12] schemes enables an archive or back-up service (prover) can retrieve a target file F, that is ,that the archive retains and reliably transmits file data sufficient for the user to recover F in its entirety. We views PORs as an important tool for semi-trusted online archives.

Erasure coding [10] can reduce the space and bandwidth overheads of redundancy in fault-tolerant data storage and delivery systems. But it introduces the fundament difficulty of ensuring that all erasure-coded fragments correspond to the same block of data. The Multiple-Replica Provable Data Possession (MR-PDP) [9] extends previous work on data possession proofs for a single copy of a file in a client server systems [1][2].Dynamic provable data possession (DPDP)[11] extends the PDP model to support provable updates to store data.

A growing number of online service providers offer to store customers 'photos, email, file system backups, and other digital assets. Currently about the risk of losing data stored with any particular service provider, reducing their incentive to rely on these services. We argue that third-party auditing [19][7][16][15][14] is important in creating an online service-oriented economy, because it allows customers to evaluate risks, and it increases the efficiency of insurance-based risk mitigation. Today, a customer must entirely trust such external services to maintain the integrity of
hosted data and return it intact. Unfortunately no service is infallible.

Cloud computing a formidable task, especially for users with constrained computing resources. Moreover, users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity. Thus, following schemes[13][19][14][15][16] enabling public audit- ability for cloud for cloud storage is of critical importance so that users can resort to a third-party auditor third-party auditor (TPA) to check the integrity of outsourced data and be worry-free. To securely introduce an efficient TPA, the auditing process should bring in no new vulnerabilities towards user data privacy, and introduce no additional online burden to user. A scheme [17] a demo of the leakage-resilient authentication and data(key) management system which can be regarded as a prominent solution for secure cloud storage. The scheme [18] ensures that eavesdroppers with access to only one of the networks are unable to decode any symbol even if they are capable of guessing some of the missing blocks. A. Jlues et al , argue[20] that Cryptography alone can't enforce the privacy demanded by common cloud computing services, even with such powerful tools as fully homomorphic encryption(FHE).

2. Cloud Data Protection Methods

2.1 Authentication and Identity

Maintaining confidentiality, integrity, and availability for data security is a function of the correct application and configuration of familiar network, system, and application security mechanisms at various levels in the cloud infrastructure. Among these mechanisms [22][21][23] are a broad range of components that implement authentication and access control. Authentication of users and even of

Manuscript received March 5, 2012
Manuscript revised March 20, 2012

communicating systems is performed by various means, but underlying each of these is cryptography.

Authentication of users takes several forms [27][29], but all are based on a combination of authentication factors: something an individual knows (such as a password), something they possess (such as a security token), or some measurable quality that is intrinsic to them (such as a fingerprint). Single factor authentication is based on only one authentication factor. Stronger authentication requires additional factors; for instance, two factor authentications is based on two authentication factors (such as a pin and a fingerprint).

Authentication is usually predicated on an underlying identity infrastructure. The most basic scheme is where account information for one or a small number Cloud Data Security: Sensitive Data Categorization 137 of users is kept in flat files that are used to verify identity and passwords, but this scheme does not scale to more than a very few systems. A full discussion of identity and access controls is beyond the scope of this paper, but the key to effective access controls is the centralization of identity. One problem with using traditional identity approaches in a cloud environment is faced when the enterprise uses multiple cloud service providers (CSPs). In such a use case, synchronizing identity information with the enterprise is not scalable. Another set of problems arises with traditional identity approaches when migrating infrastructure toward a cloud-based solution.

Infrastructure tends to employ domain-centric identity approaches that do not allow for looser alignment such as with partnership. For these reasons, federated identity management (FIM) is an effective foundation for identity in cloud computing. However, federated identity uses a claim-based token model, which entails a departure for traditional schemes. However, traditional identity needs can still be supported by a federated token model.

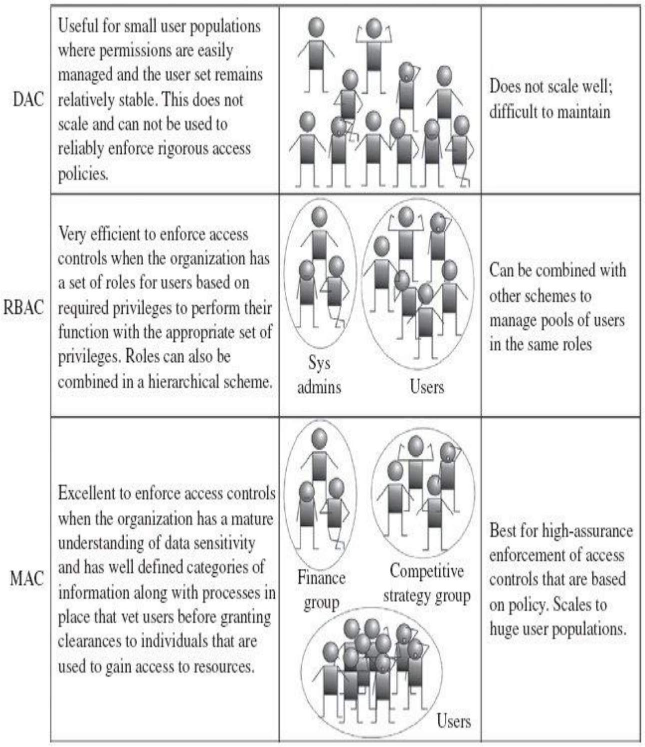
2.2 Access Control Techniques

Access control mechanisms [25][26] are a key means by which we maintain a complex IT environment that reliably supports separation and integrity of different levels or categories of information belonging to multiple parties. But access controls do not stand on their own; they are supported by many other security capabilities, access control is dependent on an identity management capability that meets the needs for the implementation.

When we discuss access controls, we refer to:

- Subjects which are people or processes acting on their behalf
- Objects such as files or other resources (a directory, device, or service of some sort)

Access controls are generally described as either discretionary or non-discretionary, and the most common access control models are:



DAC	Useful for small user populations where permissions are easily managed and the user set remains relatively stable. This does not scale and can not be used to reliably enforce rigorous access policies.		Does not scale well; difficult to maintain
RBAC	Very efficient to enforce access controls when the organization has a set of roles for users based on required privileges to perform their function with the appropriate set of privileges. Roles can also be combined in a hierarchical scheme.	Sys admins Users	Can be combined with other schemes to manage pools of users in the same roles
MAC	Excellent to enforce access controls when the organization has a mature understanding of data sensitivity and has well defined categories of information along with processes in place that vet users before granting clearances to individuals that are used to gain access to resources.	Finance group Competitive strategy group Users	Best for high-assurance enforcement of access controls that are based on policy. Scales to huge user populations.

FIGURE-1 MAC scales better for data security than other schemes do.

Discretionary Access Control (DAC). In a system, every object has an owner. With DAC, access control is determined by the owner of the object who decides who will have access and what privileges they will have. Permission management in DAC can be very difficult to maintain; furthermore, DAC does not scale well beyond small sets of users.

Role Based Access Control (RBAC) [30][31]. Access policy is determined by the system. Where with MAC access is based on subject trust or *clearance*, with RBAC [27][28][29] access is based on the role of the subject. A subject can access an object or execute a function only if their set of permissions—or role—allows it.

Mandatory Access Control (MAC. Access policy is determined by the system and is implemented by sensitivity labels, which are assigned to each subject and object. A subject's label specifies its level of trust, and an object's label specifies the level of trust that is required to access it. If a subject is to gain access to an object, the subject label must dominate—be at least as high as—the object label.

Finally, although these three access models vary in fundamental ways, they are not inherently incompatible and can be combined in different ways. As implemented, DAC generally includes a set of ownership representations (in UNIX: User, Group and Other), a set of permissions (again, in UNIX: Read, Write, Execute), and an access control list (ACL), which would list individuals and their access modes to the object, groups, and others.

Although this use of DAC may be easy to setup for a resource, as soon as there is any turnover in personnel or when the list of individuals grows, the scheme becomes unwieldy. By contrast, MAC-based enforcement scales to global user populations. Figure 5.5 depicts this point by contrasting MAC with discretionary access controls (DAC) and role-based access controls (RBAC). As shown in FIGURE-1 MAC scales better for data security than other schemes do.

2.3 Data Categorization and the Use of Data Labels

Putting in place effective and appropriate controls for information systems requires an understanding of the nature of the information. In this regard, sensitive or otherwise valuable data should be categorized to support data security. By identifying data according to sensitivity, one can implement various strategies to better protect such data. Unfortunately, understanding what other cloud data may require protection may not always be clear.

Data that a user chooses to store in the cloud may not require protection if it is not sensitive or if it can easily be recovered.

But generally, protecting data is a universal requirement regardless of its value, if for no other reason than failing to do so leads to all manner of complexity, consequence, and mischief.

In identifying and categorizing data, what we face is a multifaceted problem. Besides identifying classes of information that are sensitive or otherwise have value and labeling such information according to its characteristics, we need to protect such data, usually by means such as file permissions, encryption, or more sophisticated container approaches. We also need identity-based access controls to support organizational access policies.

Procedures are also necessary for security across phases of the data life cycle, for instance, to limit exposure of such data when we create copies or backups. Also, we need mechanisms to detect when the valuable resource is accessed in ways that warrant concern.

Data or information labeling[25][26] is one information security technique that has been used to great success for classified information such as the hierarchical categories of Unclassified, Confidential, Secret, Top Secret, and Compartmented. Labeling also supports non-classified and non-hierarchical categories such as Finance, Business Strategy, and Human Resources. The objective of information identification and categorization is to put in place an information-centric framework for controls and data handling.

SELinux and Trusted Solaris are two example operating systems that support information categorization and access enforcement for U.S. Department of Defense style mandatory access controls (MAC). Briefly, this amounts to sophisticated access enforcement by the OS and network controls.

At the heart of MAC-based security are two concepts. First, every file, discrete piece of data or network connection is marked to bound its security level with a label that the OS uses to enforce access. Second, every subject (user or process acting on behalf of a user) has a set of permissions including clearances and roles. The OS mediates all operations that subjects perform against data enforcing complex logical security operations. Although this may sound complex, and while such enforcement technology must be implemented with correctness and completeness, the concept is quite simple and the benefits enable a simplification of what otherwise would be highly complex and prone to error alternative implementations.

The Ostrich Approach (or How I Learned to Hide My Head in the Sand). In contrast to identifying sensitive data, there are many consequences when you uniformly treat all data as being equal in sensitivity or value. Without any data sensitivity oriented controls, a relatively small percentage of sensitive data is mixed in with far more non-sensitive data and is accessible to anyone with overall access. Failing to identify sensitive data complicates incident resolution and can be problematic when compromised data includes data subject to regulatory controls.

There is one misguided school of thought about this, and it can be described as the notion of hiding valuables in plain sight and hoping for the best. This is a strategy that is doomed even at the level of an individual computer used by multiple parties.

By example, one might think that credit card data can be discretely squirreled away in a file and almost impossible to locate via a search if the file system has enough files. However, such data follows defined regular patterns both in terms of the number of digits and key digits of the number. Searching for well-known strings is trivial with a computer, and because of this, several pieces of spyware do exactly that by first identifying strings such as a credit card number or a social security number and then extracting enough characters around these prizes to obtain expiration date, associated names, and along with other personal data.

Over Use of Classification. A second problem with sensitive information is a common inclination to classify or label everything as sensitive or for instance *secret*. But over classification can lead to a reduction in care in handling actually sensitive data. What we need is a balance in managing sensitive information and sound strategies for protecting the data.

2.4 Application of Encryption for Data at Rest and in Motion

Encryption [22][21][23] is a key component to protect data at rest in the cloud. Employing appropriate strength encryption is important: Strong encryption is preferable when data at rest has continuing value for an extended time period. If such long-term value encrypted data is obtained by a third party and if they have an extensive period of time to break or crack the encryption, then the reward can be well worth the effort.

There are multiple ways of encrypting data at rest. Following is an outline of various forms of encryption that serve as protection methods for securing data at rest in the cloud.

- ***Full Disk:*** Encryption of data at the disk level—the operating system, the applications in it, and the data the applications use are all encrypted simply by existing on a disk that is encrypted. This is a brute-force approach to encrypt data since everything is encrypted, but this also entails performance and reliability concerns. If encryption is not done at the drive hardware level, then it can be very taxing on a system in terms of performance. Another consideration is that even minor disk corruption can be fatal as the OS, applications, and data.

Directory Level (or Filesystem). In this use of encryption, entire data directories are encrypted or decrypted as a *container*. Access to files requires use of encryption keys. This approach can also be used to segregate data of identical sensitivity or categorization into directories that are individually encrypted with different keys.

- ***File Level:.*** Rather than encrypting an entire hard drive or even a directory, it can be more efficient to encrypt individual files.
- ***Application Level:*** The application manages encryption and decryption of application-managed data.

Critical to implementing any of these forms of encryption is the need to manage the keys that are used to encrypt and decrypt data. In addition, identifying recovery methods for when encryption keys are lost needs to be considered. When a key is lost or not available, it is important to know what options are available to recover the data for instance, do backups exist?

Also, consider the potential for side channel attacks with encryption. Simply defined, side channel attacks are attacks that target the operating nature (or environment) where the encryption is occurring in contrast to exploiting the encryption mechanisms themselves. In the context of cloud security, side channels may potentially exist by virtue of operating within the same physical infrastructure and using shared resources with other subscribers. The site sidechannelattacks.com has an extensive list of different types of side channel attacks.

Application of Encryption for Data in Motion.. The two goals of securing data in motion are preventing data from being tampered with (integrity) and ensuring that data remains confidential while it is in motion. Other than the sender and the receiver, no other party observing the data should be able to either make sense of the data or alter it. The most common way to protect data in motion is to utilize encryption combined with authentication to create a conduit in which to safely pass data to or from the cloud.

Encryption is used to assure that if there was a breach of communication integrity between the two parties that the data remains confidential. Authentication is used to assure that the parties communicating data are who they say they are. Common means of authentication themselves employ cryptography in various ways. Transferring data via programmatic means, via manual file transfer, or via a browser using HTTPS, TLS, or SSL are the typical security protocols used for this purpose. A PKI is used to authenticate the transaction (trusted root CAs), and encryption algorithms are used to protect the payload.

The following encryption Techniques are used in cloud computing.

Proxy Re-encryption [21]. A proxy re-encryption algorithm transforms cipher-text c_{k1} to cipher-text c_{k2} with a key $rk_{k1->k2}$ without revealing the corresponding clear-text, where c_{k1} and c_{k2} can only be decrypted by different key k1 and k2, respectively, and $rk_{k1->k2}$ is a re-key issued by another party, e. g., the originator of cipher-text c_{k1}.

Identity-based Encryption for Hierarchical architecture for cloud computing (HACC)[32]. In the cloud computing ,communications among the users are frequent. To achieve the secure communication , it is important to propose an encryption and signature schemes. Therefore, an Identity-Based Encryption (IBE) and Identity-Based Signature (IBS) schemes for HACC in the following.

Identity-Based Encryption is based on the above Root Private-Key Generator (PKG) setup, Lower-level setup and User-level setup algorithms. Ii is composed of encryption and decryption.

Identity-Based Signature is also based on the above Root Private-Key Generator setup, Lower-level setup and User-level setup algorithms. It incorporates two algorithms: signature and verification.

Key Policy Attribute-Based Encryption(KP-ABE). KP-ABE [33] is a public key cryptography primitive for one-to-many communications. In KP-ABE, data are associated with attributes for each of which a public key component is defined .The encryptor associates the set of attributes to the message by encrypting it with the corresponding public key components. Each user is assigned an access structure which is usually defined as an access tree over data attributes, i.e., interior nodes of the access tree are threshold gates and leaf nodes are associated with attributes. User secret key is defined to reflect the access structure so that the user is able to decrypt a ciphertext if and oly if the data attributes satisfy his access structure. A KP-ABE scheme is composed of four algorithms which are Setup, Encryption, Key Generation, Decryption.

Impediments to Encryption in the Cloud. In one example, a Software-as-a-Service public cloud, because of its very nature, might not allow subscribers to encrypt their data. This may be due to functional limitations with the actual service itself.

In the example of currently available social networks including Facebook, MySpace, and Linkedin, it is simply not possible to use encryption to ensure the confidentiality of your personal information. Nor would the cloud provider have any motivation to agree to allow this kind of data to be encrypted since many SaaS operators might not be able to provide revenue-generating services if they have an obscured view to the data they are interacting with. For instance, if Facebook was unable to intelligently interpret what kind of activities were occurring in their cloud, then how could they target you with advertisements that are most effective if they relate to your posted activities? If your data was encrypted, then that aspect of the provider's business model would be broken.

This same fact holds true to other kinds of clouds as well. IaaS providers might not be capable of encrypting at the operating system level because it would hinder their ability to monitor and therefore manage these instances.

2.5 Data Masking

Data masking is a technique that is intended to remove all identifiable and distinguishing characteristics from data in order to render it anonymous and yet still be operable. This technique is aimed at reducing the risk of exposing sensitive information. Data masking has also been known by such names as data obfuscation, de-identification, or depersonalization. These techniques are intended to preserve the privacy of records by changing the data so that actual values cannot be determined or re-engineered.

A common data masking technique involves substitution of actual data values with keys to an external lookup table that holds the actual data values. In operation, such resulting masked data values can be processed with lesser controls than if the original data was still unmasked.

But data masking must be performed carefully, or the resulting masked data can still reveal sensitive data. By example, if you mask salary data in an HR database by tokenizing what originally were employee names with name look up keys, the highest salary will probably be the CEOs. By using simple analysis techniques and methodically cross-referencing partially masked records with other employee information, more may be inferred by a process of elimination than should be.

Regardless of the masking method that is used, it is important that structures and relationships that are formed between database rows, columns, and tables are correctly maintained with each masking operation. By example, if a key to an employee table is EMP_NUMBER, changes to EMP_NUMBER must be made with identical changes in all other tables.

Correctly implemented, data masking demonstrates due diligence regarding compliance with data privacy requirements, and it can also be an effective strategy for reducing the risk of data exposure and a good strategy for countries whose privacy requirements preclude the use of cloud computing off-EU territory for privacy information.

3. Conclusion

Security in the cloud must be included from the start. This demands a new approach to end-to-end security that support strong isolation of data, even when business process are outsourced into the cloud. Cloud processing needs isolation between users of shared services, as well as isolation between services.

The main issue with KP-ABE is that it would introduce a heavy- computation overhead for the data owner to re-encrypt data files and might require the data owner to be always online to provide secret key update service for users. To resolve this issues, we combine the technique of proxy re-encryption with KP-ABE and delegate tasks of data to cloud servers. Confidentiality of user access privilege and user secret key accountability can be achieved.

Proxy re-encryption pairing-based schemes realize important new features, such as safeguarding the master secret key of the delegator from a colluding proxy and delegatee. Secure storage system is an important application of proxy re-encryption.

The driving motivation for role-based access control (RBAC) is to simplify security policy administration while facilitating the definition of flexible customized policies. Today, RBAC is coming to be expected among large organization and the number of vender that offer RBAC features is growing rapidly. RBAC models have been shown to be “policy-neutral” in the sense that by using role hierarchies and constraints, a wide range of security policies can be expressed Security administration is also greatly simplified by the use of roles to organize access privileges.

References

[1] G. Atenniese, R.Burns, R.Curtmola, J. Herrring, L. Kissner, Z. Peterson, and D. Song. Provable data possession at untrusted stores, In Proc. ACM CCS.Pages 598-609,2007

[2] G. Atenniese, R.Dipietro,L.V. Mancini, and G.Tsudik. Scalable and efficient provable data possession,2008. IACR ePrint manuscript 2008/114.

[3] K. Bowers, A. Juels, and A. Oprea. Proofs of retrievability: Theory and implementation,2008. Available from ePrint, report 2008/175.

[4] Y. Dodis, S. Vadhan, and D. Wichs. Proof of retrievability via hardness amplification. In TCC,2009.

[5] A. Juels and B. Kaliski, PORs: Proofs of retrievability for large files. In Proc. ACM CCS, pages 584-597,2007.

[6] Hovav Shancham and Brent waters. Compact proofs of retrievability. In proc. ASIACRYPT'08,pages 90-107,Berlin, Heidelberg,2008.Springer-Verlag.

[7] M.A. Shah, M. Baker, J.C. Mogul, and R.Swaminathan. Auditing to keep online storage services honest,2007. Presented at HotOS XI, May 2007.

[8] M. Shah, R. Swaminathan, and M. Baker. Privacy-preserving audit and extraction of digital contents. Cryptology ePrint archive, April 2008. Report 2008/186.

[9] R. Curtola O. Khan, R. Burns, and G. Ateniese. MR-PDP: multiple –replica provable data possession . In proc. Of the 28th IEEE International Conference on Distributed Computing Systems (ICDCS'08),2008, to appear.

[10] J. Hendricks, G.R. Ganger ,and M.K. Reiter. Verifying distributed erasure-coded data. In 26th ACM Symposium on Principles of Distributed Computing (PODC),2007.

[11] C. Erway, C.Papamanthou, A. Kupcu and R. Tamassia. Dynamic provable data possession. In ACM Conf. on Computer and Communications Security 2009 (to appear) . Available as Cryptology ePrint Archive, Report 2008/ 432.

[12] K.D. Bowers , A. Juels. and A.Oprea, " HAIL: A high-availability and integrity layer for cloud storage," in proc. Of CCS'09,2009, pp. 187-198.

[13] M.A. Shah ,R. Swaminathan ,and M. Baker,"Privacy-Preserving audit and extraction of digital contents," , Cryptology ePrint Archive ,Report 2008/186, 2008, http://eprint.iacr.org/.

[14] Q. Wang, C. Wang, J. Li, K. Ren, and W.Lou," Enabling public verifiability and data dynamics for storage security in cloud computing," in proc. of ESORICS '09, Volume 5789 of LNCS.Springer-Verlag, Sep. 2009, pp. 355-370.

[15] C. Wang, Q.Wang, K.Ren, and W.Lou," Privacy-Preserving public auditing for storage security in cloud computing," in proc. of IEEE INFOCOMM ' 10,San Diego, CA, USA, March2010.

[16] C. Wang ,K.Ren, W.Lou, and J.Li," Towards publicly auditable secure cloud data storage services," IEEEE Network magazine , Vol.24, no.4, pp.19-24,Dec 2010.

[17] S.H. Shin ,K.kobara," Towards secure cloud storage," Demo for CloudCom2010, Dec 2010.

[18] P.F. Oliveira, L.Lima, T.T.V. Vinhoza, J. Barros, M. Medard," Trusted storage over intrusted networks," IEEE GLOBECOM 2010, Miami, FL. USA.

[19] M.A. Shah et al.," Auditing to keep online storage services honest," proc. USENIX HotOS '07,May 2007.

[20] M. Dijk, A.Juels," On the Impossibility of Cryptography Alone for Privacy- Preserving Cloud Computing," HotSec 2010.

[21] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger. Improved Proxy Re-encryption Schemes with Applications to Secure Distributed Storage. ACM Trans. Inf. Syst. Secure., 9:1-30, February 2006.

[22] Saman Zarandioon, Danfeng Yao, and Vinod Ganapathy. K2C: Cryptographic Cloud Storage with Lazy Revocation and Anonymous Access. In proceeding of securecomm, 2011.

[23] H. Xiong, X. Zhang, W. Zhu, and D. Yao. Cloudseal: End-to-End Content Protection in Cloud-based Storage and Delivery Services. In proceeding of securecomm,2011.

[24] Y.P. Chiu, C.L. Lei, and C.Y. Haung. Secure Multicast Using Proxy Encryption. In information and communication ssecurity, Lecture Notes in computer science, 2005.

[25] V. Goyal , O. Pandey, A. Sahai, and B. Waters. Attribute-based Encryption for Fine-grained Access Control of Encrypted Data. In proc. of ACM CCS,2006.

[26] S. Yu, C. Wang, K. Ren, and W. Lou. Achieving Secure, Scalable ,and Fine-grained Data Access Control in Cloud Computing . In INFOCOM '10.

[27] L. Zhou, V. Varadharajan, and M. Hitchens. Enforcing role-based access control for secure data storage in the cloud. The computer Journal, 2011.

[28] M.L. Daimani, E. Bertino, B. Catania, and P. Perlasca, "GEO-RBAC: A spatially aware RBAC," in ACM Transactions on Information and System Security (TISSEC),2007.

[29] F. Hansen and V. Oleschuk," SRBAC: A spatial role-based access control model for mobile systems," in proc of 8th Nordic Workshop on Secure IT Systems (NORDSEC), October 2003.

[30] American National Standard for Information Technology: Role-based access control. ANSI INCITS 359-2004(2004).

[31] Bacon, J., Moody, K.,Yaom W. :A model of OASIS role-based access control and its support for security. ACM Transactions on Information and System Security (TISSEC) 5(4) (November 2002) 492-540.

[32] A. Boldyreva, V. Goyal, and V. kumar. Identity-based encryption with efficient revocation. In WOSN,2008.

[33] V. Goyal, O. Pandey, A. Sahai, B. Waters," Attributes-based encryption for fine-grained access control of encrypted data," in Proc. of CCS'06,2006.

Satyendra Singh Rawat received the Bachelor of Engineering in ComputerScience from Madhav Institute of Technology & Science, Gwalior, M.p., India. In 2005.M.Tech (CSE) Research Scholar, RKDF Institute of Science & Technology, Bhopal, M. P., India.