Scenarios for Leveraging Predicate Based Encryption (PBE) in the Cloud

[1]S. Neelima, [2]M. Padmavathi, [3]Y. Lakshmi Prasanna
[1,2]Dept. of MCA, Swarna Bharathi Institute of Science and Technology, Khammam, AP, India
[3]Dept. of IT, Swarna Bharathi Institute of Science and Technology, Khammam, AP, India

Abstract

Cloud computing has become one of the most significant data security issue in recent years. That is due to the dramatically emerging applications and required services of cloud computing. However, in order to safely utilize and enjoy the benefit of cloud computing through wired/wireless networking, sufficient assurance of data security such as confidentiality, authentication, nonrepudiation, and integrity is the most critical factor for adoption. Data that was once housed under the security domain of the service user has now been placed under the protection of the service provider. Users have lost control over the protection of their data. No longer is our data kept under our own watchful eyes. This paper explains how Predicate Based Encryption (PBE) could be leveraged within the Cloud to protect data. Five scenarios for using Predicate Based Encryption within the Cloud are presented. These scenarios differ in terms of mode of operation, predicate placement, and ownership over the Key Authority.

Keywords

Cloud Computing, Cloud Security, Predicate Based Encryption

I. Introduction

Cloud Computing is the name given to a recent trend in computing service provision. This trend has seen the technological and cultural shift of computing service provision from being provided locally to being provided remotely and en masse, by third-party service providers. These third-parties offer consumers an affordable and flexible computing service that consumers would otherwise not have been accessible, let alone afford. This new means of service provision has evolved from and is the culmination of research stemming from (among others) distributed and networked systems, utility computing, the web and software services research. This paradigm shift has led to computing being seen as another household utility, aka “fifth utility”, and has prompted many a business and individual to migrate parts of their IT infrastructure to the cloud and for this data to become managed and hosted by Cloud Service Providers (CSPs). However, Cloud Computing is the cause célèbre among tech pundits and has led to the term `Cloud Computing’ as an umbrella term being applied to differing situations and their solutions. As such a broad range of definitions for Cloud Computing exists, each of which differ depending on the originating authors’ leaning.

Predicate Based Encryption (PBE), represents a family of asymmetric encryption schemes that allows for selective fine-grained access control as part of the underlying cryptographic operation. The origins of PBE are in Identity Based Encryption (IBE). In IBE schemes an entity’s encryption key is derived from a simple string that represents the entity’s own public identity e.g. an email address. PBE schemes offer a richer scheme in which an entity’s `identity’ can be constructed from a set of attributes and decryption is associated with access policies that offers a more expressive means with which to describe the relation between the attributes. Various constructions of PBE schemes have been proposed that use general predicates (represented as Boolean formula) or specific predicates such as equality, hidden vector or inner product. The choice of predicate will have a direct affect upon the scheme, its characteristics and the composition of the access policies. Moreover, the placement of the predicate (either with the cipher-text or the decryption key) has a great affect upon the workings of a PBE scheme.

The focus of this thesis now turns towards how Predicate Based Encryption (PBE) can be used within the cloud. The Public Key Infrastructure (PKI) mode allows a third-party to offer other entities a means through which they (the other entities) can share their data. The Multicast Content Provision (MCP) mode allows an entity the means through which they can selectively share their data with others without having to rely upon a third-party. The final mode, that of Duty Delegation

Infrastructure (DDI), allows an entity to restrict access to data that has been sent to them. However, the application of these three modes of operation to a Cloud setting will be affected by:

- Whether the Key Authority (KA), the owner of the PBE scheme is a service user or a service provider i.e., Cloud Service Provider (CSP)
- The style of access control required i.e. Cipher text-Policy (CP) or Key-Policy (KP)

II. Scenarios for using Predicate Based Encryption (PBE)

From the above factors, five viable scenarios emerged:

A. Scenario I: Embedded within a Service

A CSP incorporates a CP-PBE scheme in PKI mode within an existing service. Providing service users the means with which they can share data amongst each other within the domain of a particular service.

B. Scenario II: PBE-as-a-Service

Similar to Scenario I with the exception that PBE is used out with the confines of a particular service domain.

C. Scenario III: 'Database Submission’

A CSP offers a service in which service users can submit data to a database. A KP-PBE scheme in DDI mode is used to restrict access over the submitted content to authorized employees of the CSP and other affiliated parties.

D. Scenario IV: ‘Database Access’

The antithesis of Scenario III. A CSP offers content that can only be accessed by subscribed users. A KP-PBE scheme is used in MCP mode to restrict subscribed users’ access to the CSP’s content.

E. Scenario V: 'Distributed Security’

The service user deploys their own CP-PBE scheme in MCP mode. PBE is used to mitigate access over data the user pushes to the Cloud.

III. Leveraging PKI mode

The PKI mode of operation leans naturally towards the as-a-Service paradigm, and consequently its use by a CSP. The CSP is the third-party offering the use of PBE to its service users. With this in mind, PBE can be offered by the CSP in two distinct ways, either: a) embedded within a service; or b) provided as a service. In both cases, the CSP is promoting the sharing of data amongst other users of the services; sharing is a dominate feature exhibited by many Software as a Service (SaaS) services. Leading to the implication that these two scenarios, at least, appear to be best suited for SaaS services. Moreover, an access control style is required that allows the encrypting entity to state explicitly for whom access will be granted on a per message basis. Cipher text-Policy schemes provide such access control. Indicating that the attribute universe is used to, at least, describe users of the service.

A. Scenario I: Embedded within a Service

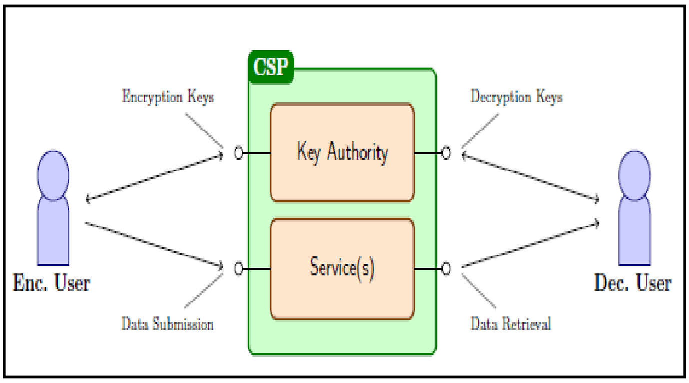

Fig. 1: Outline of Scenario I

Embedding a CP-PBE scheme within an existing service allows for the complete integration of PBE within the service itself. Fig. 1 outlines the core interactions within this scenario. Upon registering with the service, new users are provided with a decryption key derived from a set of attributes that describe the user. When pushing data to the service, service users encrypt their data under access policies of their choosing. Moreover, as the KA is embedded within the service, the CSP can also aid in policy administration. In other words, the CSP can offer service users a means through which to specify policy rules and thus construct access policies.

Due to embedding PBE within the service itself, the attribute universe can be extended to refer to service specific functionality such as the CSP's ability to read the users data for targeted ads. For example, take an Online Social Network (OSN). The attribute universe can identify service users through attributes such as: identification number, gender, school networks, interests, location, D.O.B. et cetera. Functionality such as, the visibility of users' messages can also be included within the same attribute universe. Following on from indicating the visibility of a user's message is the notion that the CSP can also be referenced within the policy rules and treated as a user in their own right. Though this does not adhere strictly to the PKI's modus operandi it nonetheless allows the encrypting user to explicitly opt-in the CSP when the user constructs policy rules. Moreover, such an attribute can be combined with a date attribute to provide the CSP with a limited window of opportunity to access encrypted data. This is a powerful construct, the user has explicitly stated within an access policy that the CSP can access the encrypted data, and more importantly for how long.

A problem for this scenario is that service users need to trust the CSP. As the CSP is the KA, the CSP could easily construct their own arbitrary decryption keys, and use these keys to decrypt service users' messages. Service users still need to trust the CSP to not be malicious in this respect. This trust can be enforced through legal means i.e. privacy policies and service level agreements. Furthermore, using PBE as described requires that each CSP provide their own scheme. With each service that the user interacts with a different PBE scheme needs to be taken into account. That is, separate key management facilities and algorithms need to be managed by the user; one per service. Given what is known over key storage issues this could be a problem. Although, how the PBE is actually deployed will affect this somewhat.

B. Scenario II: PBE-as-a-Service

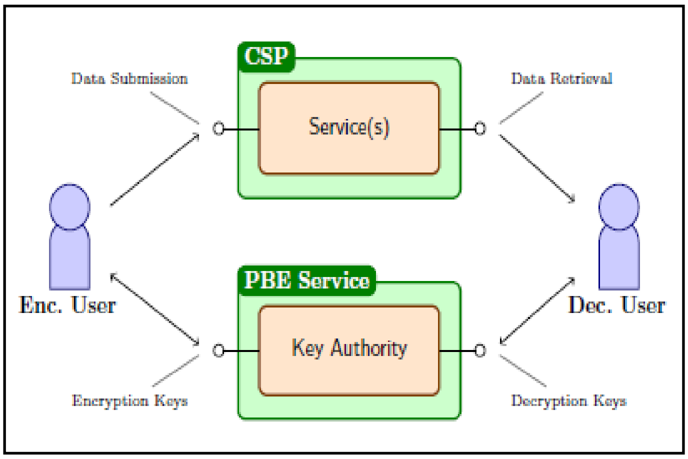

Fig. 2: Outline of Scenario II

Providing a CP-PBE scheme as a service in itself allows service users to interact with multiple services, offered by different CSPs, and use a common solution. This reduces the overhead concerning key management, and also the different encryption and decryption algorithms that a service user must be aware of. Providing `PBE-as-a-Service' makes this scenario an example of Security-as-a-Service, more specifically it can be tied directly into Identity-as-a-Service services. This implies that not only can this scenario be used to protect SaaS level services but also at the Platform as a Service (PaaS) level as a service. Fig. 2, outlines the core interactions within this scenario. Like Scenario I, users possess decryption keys comprised of a set of attributes used to describe the user themselves. When interacting with different services the user can push encrypted data knowing that only others who can satisfy the access policy can access the underlying plain-text data. Policy administration service can also be offered by the PBE service. As with Scenario I CSPs should be treated as users in their own right. Hence, when constructing policy rules the encrypting user can explicitly opt-in the CSPs as someone with whom they wish to share their data with. Moreover, CSPs `service users' can themselves integrate this pre-existing PBE service into their own service offerings. Unlike Scenario I, however, service users can treat the CSP like any other user and not have to worry about the CSP's maliciousness. A CSP in this setting is unable to construct arbitrary decryption keys. Users can use a service even though they may have no trust in the CSP offering that service. When the user does have trust in the CSP to access the user's data, then the CSP can be opted in. However, the service agnosticism does present several interesting issues. For one, the attribute universe is defined by a third-party who may not possess knowledge of service specific information, offering users attributes that describe information out with the context of a service. This may affect the expressiveness of policy rules as defined by encrypting users. Offering a PBE service does reduce the overhead in terms of key management and knowledge of algorithms, however, this reduction in overhead comes at some cost. Scenario I is highly compartmentalized, CSPs are only able to access the data that has been pushed to their service. The PBE service provider, if malicious, has the ability to construct decryption keys to access data that a user has pushed to multiple services.

IV. Leveraging DDI Mode

A. Scenario III: “Database Submission”

The rationale behind the DDI mode is controlling access over submitted data. When a service user uses this mode of operation it implies that the service user themselves will be controlling access to data that has been submitted to them. This has no obvious benefits or uses. It is better for the user to utilize the MCP mode of operation. On the other hand with a CSP as the KA, an interesting use case does emerge. With DDI mode the CSP is using PBE to facilitate selective access over data, submitted expressly for use by the CSP, under the proviso that said data will only be accessible by the CSP’s own employees and affiliates. That is data is submitted by service users and the CSP allows authorized entities access to select subsets of all data that has been submitted. This essentially describes the operation of a database. Databases can be thought of in terms of: a) who is submitting the data; b) who is accessing the data; and c) the data itself. With DDI mode, the database is stored with the CSP. Data is submitted for the CSP by service users, and is being accessed by the CSP’s employees and other affiliates. Fig. 3. Outlines the core interactions for this scenario.

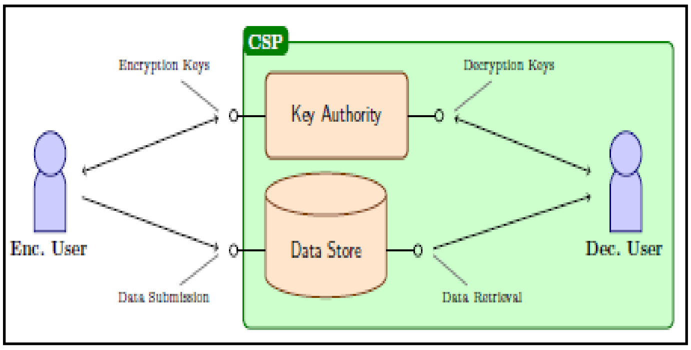

Fig. 3: Outline of Scenario III

Within databases one needs to control access not only to the database itself but also to the data stored within. Obviously, users’ access over the data within the database needs to be proscriptive and user’s ability to access individual records needs to be curtailed. Key-Policy schemes provide such access control. As such data is encrypted under a set of attributes, and decryption keys from a predicate. Another and more intuitive reason for using Key-Policy schemes stems from the decryption keys. Decryption keys in this setting represent simple search queries. When submitting data, service users encrypt their data under a set of attributes. The CSP can then internally determine on a per employee basis what the employee can access from this database. Moreover, with numerical attributes it is feasible to limit the period during which the employee can use their query.

The analogy of a `database’ implies the deployment of PBE as part of either a SaaS, or PaaS service. For example, a course submission service could use DDI mode to control submission of student’s submission. During submission, students could encrypt their course under a set of attributes that describe1:

- The student’s matriculation number
- The course code
- The assignment number
- Submission time. Lecturers and teaching assistants can then be assigned decryption keys pertaining to the course(s) they are associated with.

V. Leveraging MCP Mode

MCP mode of operation an encrypting entity is the Key Authority (KA). Within a cloud setting two scenarios emerge when: a) a service user is the encrypting entity; and when b) a CSP is the encrypting entity. With a service user Cipher text-Policy schemes are better suited; access needs to be decided per message. With the CSP as the encrypting entity the setting is more data centric and as such a Key-Policy scheme will suffice.

A. Scenario IV: 'Database Access'

CSPs can utilize MCP mode to restrict access to the content that they produce in much the same manner as was seen with Scenario III. The difference lies in the involved actors and origin of the data. Fig. 4, outlines the actors involved and core interactions between them.

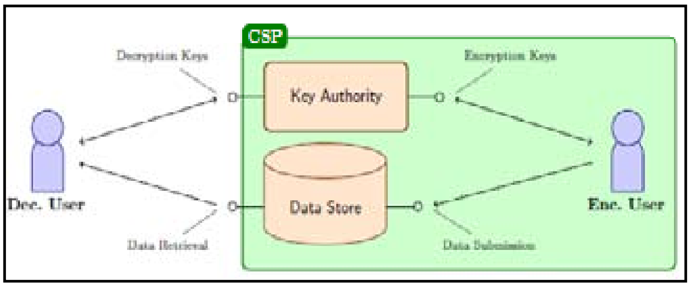

Fig. 4: Outline of Scenario IV

Employees of the CSP together with authorized affiliates are those permitted to encrypt data. Decryption of data is performed by service users. When signing up to a service, users are assigned a decryption key derived from some policy rule that describes the data they are allowed to access. Such policy rules can be used to indicate service related information such as subscription level and periods of time. This provides the CSP with a means to enforce different subscription models i.e. fermium. With MCP mode the act of restricting access can also extend to any content streams that the CSP produces and broadcasts. This implies that this scenario can be used at both the PaaS and SaaS service levels. Take a media streaming service such as Natick’s, streamed content can be encrypted under a set of attributes that describe, for example: a) the content’s rating i.e. U, PG, 12, 16, 18; b) the subscription level e.g. free, gold member, silver member; and c) the content’s details such as title, season and episode.

B. Scenario V: 'Distributed Security'

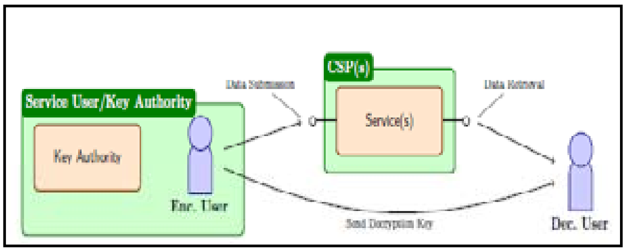

Fig. 5: Outline of Scenario V

Scenarios I and II present solutions to protecting data they wish to share via the cloud. The underlying problem with these solutions is the reliance upon a third-party to act as the KA. When deployed by a service user the MCP mode presents an improvement with respect to trusting a third-party. With the MCP mode there is no need for a trusted third-party over key management; users provide their own PBE scheme.

When signing up to a service or when starting to interact with a new `friend', the encrypting user i.e. the Key Authority, will create and assign to the entity a decryption key. This decryption key will be derived from a set of attributes that describes how the decrypting user relates to the encrypting user. Through use of a Cipher text-Policy scheme, the encrypting user can encrypt their data locally under a policy rule, and then push the cipher-text into the cloud. Attributes can be used to indicate, for example: a) type of relationship with another entity i.e. an associate, a friend, a close friend, a really close friend, family, or CSP; b) visibility of message i.e. is the message to be private or public; and c) an identifier for each domain that the user has interacted with i.e. on an OSN. This use of the MCP mode of operation presents a decentralized solution to the problems associated with Scenarios I and II. Implying this scenarios use at the PaaS and SaaS service levels, with each `user' providing their own encryption scheme. From this two interesting observations can be made that can affect the management and deployment of this mode. First, the encrypting user is responsible for the remit of the Key Authority. The user is responsible for policy rule administration, responsible for the correct assignment of decryption keys, responsible for key management et cetera. Secondly, the security guarantees provided cover unidirectional communication. Each user utilizes PBE to protect their own data only. For n users communicating it requires each user to take into account the decryption keys, and associated decryption algorithms for n - 1 other PBE schemes.

VI. Comparing Scenarios

The five different scenarios can be divided into two distinct groupings concerning the style of interaction (between the service user and CSP) that the scenarios are used to protect: bidirectional and unidirectional. This grouping can be used to compare related scenarios. Moreover, a good solution for protecting the confidentiality of data pushed to the cloud should take into account the following three points:

- User empowerment does not alone equal Data Obfuscation.
- The user need not be responsible for ensuring all security guarantees.
- Some CSPs may be evil but some are more evil than others.

A. Comparing Scenarios I, II and V

Scenarios I, II and V describe a means to protect the results of a bidirectional communication style. With this style of communication a service user expects to push their, and pull other service users' data from the cloud. PBE is used to ensure confidentiality of this data.

1. User Empowerment

In each of the three scenarios, service users have control over who precisely can decrypt their data, and by extension how it can be used. They have become empowered. However, the degree of empowerment depends upon the KA. The KA is responsible for defining the attribute universe, and assignment of decryption keys. This in turn affects the expressiveness of access policies in terms of what attributes the user can specify, and the effect of application e.g. attributes that identify if the CSP can decrypt encrypted data. Scenario I offer tight integration with a single service. The CSP can design an attribute universe that caters to the needs of the service. On the other hand, Scenario II does not offer as tight integration. The definition of the attribute universe is dependent upon a third-party who is not necessarily aware of a particular services needs. Finally, Scenario V offers the service user complete freedom over the composition of the attribute universe, and moreover, the service user is also in control of who is able to obtain decryption keys.

2. User Responsibilities

With these scenarios, the CSP is not afforded complete trust in their abilities to secure user's data. In Scenario V the service user has no trust in the abilities of the CSP to offer data confidentiality. The service user is responsible for all matters, excluding decryption, arising from the use of the PBE scheme. Scenario I offers a solution that allows the service user to have some trust in the CSP. The CSP is responsible for all matters excluding the specification of encryption keys; this is left for the service user. Finally, Scenario II implies that the service use has no trust in the CSP to protect the confidentiality of users' data. However, with Scenario II this service user must now have some trust in the ability of the `PBE Service' to fulfill its remit.

3. Trust in the CSP

Each of the three scenarios offers differing levels of trust over the CSP. The preceding paragraph explained this in relation to who provides assurances over data confidentiality. Establishment of the malicious CSP acting as the KA will be able to construct arbitrary decryption keys. This is not a problem for Scenario V. For all three scenarios, however, service users can explicitly stipulate on a per message basis if the CSP can access the service users data.

B. Comparing Scenarios III and IV

Scenarios III and IV, describe a means to protect a distinctly unidirectional style of interaction. With Scenario III, the service user is pushing data to the cloud. With, Scenario IV the service user is pulling CSP's data from the cloud. These scenarios do not necessarily seek to empower service users over the confidentiality of their data: They seek to empower the CSP instead. Moreover, both of these scenarios allow the CSP to provide a form keyword search using expressive queries over an `encrypted database'.

Scenario III presents a scenario in which the user is afforded some control over access to their data through specification of the encryption key. Ultimately, however, it is the CSP that has been empowered. This solution presents users with a solution that they must have complete trust in the CSP to control access to the submitted data.

With Scenario IV, the three guidelines iterated at the start of this section cannot be applied. This scenario seeks to empower the CSP over access to their data, and not service users. In this scenario the CSP is in control over the attribute universe and over issuing decryption keys. The CSP has guarantees over the confidentiality of their content, and allows them to enforce not only fine grained access control but also support for tiered content provision.

VII. Conclusion

Cloud Computing embodies the as-a-Service paradigm and allows for services to be provided en masse to consumers. When combined with the cloud setting, two different sets of scenarios emerged based upon whether the service user's or CSP's data was to be protected. PBE schemes can be used to protect service user's data in three different scenarios: Scenario I saw the inclusion of PBE within a service; Scenario II saw the provision of PBE as-a-Service; and Scenario V saw PBE being deployed by the user themselves. In each of these three scenarios PBE can be used by service users to specify precisely with whom they wish to share their data, for what purpose, and for how long. Although Scenario V may be a privacy zealot's ideal choice , they are in full control ,its practical feasibility has yet to be determined.

The remaining two scenarios, on the other hand, do appear to be more promising. However, these scenarios in themselves do present a dilemma between usability and the guarantees made over end-to-end security. When looking to protect CSP's data, PBE can facilitate keyword search with complex queries over encrypted data: Scenario III by the CSP; and in Scenario IV by a service user. This use of PBE is rather interesting in that the focus of these scenarios is on the CSP and not service user, and is most certainly worthy of further investigation. The use of PBE within the cloud appears to be concentrated at both the PaaS and SaaS service layers. In summation, the scenarios presented in this paper do show that the use of PBE in the Cloud is advantageous, moreover recent papers supports these claims.

VIII. Acknowledgement

The authors would like to acknowledge and express their heartfelt gratitude to the Swarna Bharathi Institute of Science and Technology [SBIT] Management, Chairman, Director, Principal and Dean R&D. for their vital encouragement, support and facilities providing by them to achieve this task and at they would like to thank Anonymous Reviewers for their valuable suggestions and comments. This paper has greatly benefited from their Efforts.

References

[1] Michael Armbrust, Armando Fox et al. (2009),"Above the Clouds: A Berkeley View of Cloud Computing", Tech. rep. UCB/EECS-2009-28. Electrical Engineering and Computer Sciences, University of California at Berkeley, [Online] Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS- 2009-28.html.

[2] G. Alp_ar, J.-H. Hoepman, J. Siljee,"The Identity Crisis. Security, Privacyand Usability Issues in Identity Management".

[3] Ken Birman, Gregory Chockler, Robbert van Renesse, "Toward a cloud computing research agenda", SIGACT News 40.2 (2009), pp. 68-80, [Online] Available: http://doi.acm.org/10.1145/1556154.1556172.

[4] Philippa J. Broadfoot, Andrew P. Martin,"A Critical Survey of Grid Security Requirements and Technologies", Tech. rep. PRG-RR-03-15. Wolfson Building Oarks Road Oxford OX1 3QD: Oxford University Computing Laboratory, 2003, [Online] Available: http://www.comlab.ox.ac.uk/files/930/RR-03-15.ps.gz.

[5] Monica Chew, Dirk Balfanz, Ben Laurie,"(Under) mining Privacy in Social Networks", In: Proceedings of the 2008 IEEE Symposium on Security and Privacy Workshop: Web 2.0 Security and Privacy - W2SP 2008. Publish Online. 2008, [Online] Available: http://w2spconf.com/2008/papers/s3p2.pdf.

[6] Flavin Cristian,"Understanding fault-tolerant distributed systems", In: Commun. ACM 34.2 (1991), pp. 56-78. [Online] Available: http:// doi.acm.org/10.1145/102792.102801.

[7] Gabriele D'Angelo, Fabio Vitali, Stefano Zacchirolo. "Content Cloacking: Preserving Privacy with Google Docs and other Web Applications", To appear in the 25th Symposium on Applied Computing (SAC'10), March 22-26, 2010, Sierre Switzerland. Mar. 2010.

[8] Cloud Computing: Benefits, risks and recommendations for information security. Tech. rep. European Network and Information Security Agency (ENISA)2009, [Online] Available: http://www.enisa.europa.eu/act/rm/files/deliverables/cloud-computing-risk-assessment.

[9] Brian Hayes,"Cloud Computing", Commun. ACM 51.7 (2008), pp. 9-11, [Online] Available: http://doi.acm.org/10.1145/1364782.1364786.

[10] Meiko Jensen, Jorg Schwenk et al.,"On Technical Security Issues in Cloud Computing", In: Cloud Computing, IEEE International Conference on 0(2009), pp. 109{116. doi: [Online] Available: http://doi.ieeecomputersociety.org/10.1109/CLOUD.2009.60.

[11] Graham Kirby, Alan Dearle et al.,"An Approach to Ad hoc Cloud Computing", Tech. rep. St Andrews Cloud Computing Initiative, School of Computer Science, University of St Andrews, Feb. 2009, [Online] Available: http://arxiv.org/abs/1002.4738.

[12] Michael McIntosh, Paula Austel,"XML signature element wrapping attacks and countermeasures", SWS '05: Proceedings of the 2005 workshop on Secure web services. Fairfax, VA, USA: ACM, 2005, pp. 20-27, [Online] Available: http://doi.acm.org/10.1145/1103022.1103026.

[13] Tim Mather, Subra Kumaraswamy, Shahed Latif,"Cloud Security and Privacy: An Enterprise Perspective on Risk and Compliance", Editor Mike Loukides. O'Reilly, 2009.

[14] Siani Pearson,"Taking account of privacy when designing cloud computing services", In: CLOUD '09: Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing. Washington, DC, USA: IEEE Computer Society, 2009, pp. 44{52. [Online] Available: http://dx.doi.org/10.1109/CLOUD.2009.5071532.

[15] Malden A. Vouk."Cloud Computing, Issues, Research and Implementations", In: Journal of Computing and Information Technology 16 (2008), pp. 235, [Online] Available: http://cit.srce.unizg.ur/index.php/CIT/article/view/1674/1378.

[16] Luis M. Vaquero, Luis Rodero-Merino et al.,"A break in the clouds: towards a cloud definition", In: SIGCOMM Comput. Commun. Rev. 39.1 (2009), pp. 50-55. [Online] Available: http://doi.acm.org/10.1145/1496091. 1496100.

[17] Murat Kantarcioglu, Ahmed K. Elmagarmid, Amit P. Sheth,"A Survey of Privacy-Preserving Methods Across Horizontally Partitioned Data", Privacy-Preserving Data Mining, Vol. 34. Advances in Database Systems. Springer US, 2008, pp. 313-335, [Online] Available: http://dx.doi.org/10.1007/978-0-387-70992-5_13.

Mrs. S. Neelima, working as an Associate professor in the department of Master of Computer Applications. Her research areas include Cloud Computing, Data Ware housing and data mining, Computer Networks and Network Security.

Mrs.M.Padmavathi, working as an Associate professor in the Department of Master of Computer Applications. Her research areas include Cloud Computing, Mobile Computing, Data Ware housing and data mining and Image Processing.

Mrs. Y. Lakshmi Prasanna, working as an Associate professor in the department of Information Technology. Her research areas include Cloud Computing, Computer Networks, Mobile Computing and Data Warehousing and data mining.