International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 10 | Oct 2019 www.irjet.net p-ISSN: 2395-0072

Model-Driven Platform for Service Security and Framework for Data Security and Privacy Using Key Management in Cloud Computing

Y. Kiran Kumar[1], Dr. R. Mahammad Shafi[2]

[1]*Research Scholar, Department of Computer Science, Bharathiar University, Coimbatore, T.N., INDIA*
[2]*Research Supervisor, Department of Computer Science, Bharathiar University, Coimbatore, T.N., INDIA*

Abstract - *Cloud computing is a technology to share the data and resources used among various organizations, but the security and privacy are most important aspects of cloud computing. The main responsibility of cloud service provider is the quality of service. Many of the cloud computing security frameworks have faced many challenges in security that has not yet been addressed well. The data accessed and shared through many devices from the cloud environment are not secure because they are likely to have various attacks like Identity Access Management (IAM), hijacking an account or a service either by internal/external intruders. In this paper, we proposed a model-driven approach enables the definition of security requirements at the modeling layer and facilitates a transformation based on security configuration patterns. We have also proposed a new system which can prevent the exposure of the key as well as a framework for sharing a file that will ensure security using key management system within the cloud environment and also suggests to the approach of Multi-layer Encryption techniques in cloud computing thus enhancing the security parameters concerning sensitive data. Thus with layer ways Encryption technique the data in cloud server can be made more secured with better privacy. Resultant both cloud side and owner of the data gain enhanced security. According to this encryption technique if data-owner's authorization is not granted then the users are restricted from the data access.*

Key Words: Service Oriented Architecture, Cloud Computing, Encryption, Access Control, Multi-layer encryption, Privacy, Security.

1. INTRODUCTION

Cloud computing is a relatively new business model for outsourced services. However, the technology behind cloud computing is not entirely new. Virtualization, data outsourcing, and remote computation have been developed over the last 20 years, and cloud computing provides a streamlined way of provisioning and delivering such services to customers. In this regard, cloud computing has often been criticized as representing just a new trend, rather than an innovative computing technology. As such, it is often best described as a business paradigm or computing model rather than any specific technology. A cloud consumer adopting a cloud-based solution needs to follow these steps:

1. Describe the service or application for which a cloud-based solution may be leveraged

2. Identify all functional capabilities that must be implemented for this service

3. Identify the security and privacy requirements and the security controls needed to secure the service or application.

The trust relation between cloud customers (CCs) and cloud service providers (CSPs) has to be established before CCs move their information systems to the cloud. This requires an in-depth understanding of associated risks. Moreover, regulations related to data protection, financial reporting, etc. involve certain requirements that should be complied with when outsourcing business processes to third parties, like CSPs. User authentication and authorization among cloud actors is a critical element of cloud architecture. Without knowing who is logging into the cloud-based information system, and who is accessing what data, cloud actors are not able to protect the data housed by a cloud ecosystem. Understanding who the users are, what data they are trying to access, where the data are stored, and how are users trying to get to these data—these are critical pieces of information that help cloud consumers determine an appropriate cloud architecture and deployment model.

1.1 Preliminary

We assume that the cryptographic algorithms to encrypt data are secure. Meanwhile, we assume that the random string will not be repeatedly generated by the clients. We assume that the client of each collaborator runs in a secure environment which guarantees that:

• The generation and distribution of shared secret and privilege management on the client of the initiator are appropriately maintained.

• The secret passcode and keys that appear in the clients would not be stolen by any attackers.

• The communication channel between the client and the cloud is enough to transmit all necessary data in real time and protected by existing techniques such as SSL/TLS.

1.2 Access Control

Traditional access control architectures are based on the assumption that data storage management is located within a trusted domain and the owner has adequate knowledge about the system. However, this assumption is no longer valid in the cloud computing paradigm. Multiple

stakeholders are engaged as users within the cloud platform and have different levels of data access permission. As a result, a greater granularity of access control is required to ensure that each stakeholder has access to exactly what they are authorized and to ensure the privacy and confidentiality of the cloud-based services. Researchers and experts are mostly concerned about outside attackers when considering the security issues in distributed systems. Therefore, significant efforts have been made to keep the malicious attacker outside of the perimeter. Unfortunately, such efforts cannot always be effective in the cloud computing paradigm. The incident where Google fired engineers for breaking internal privacy policies confirms that attackers may reside within the service framework [1]. Carnegie Mellon University's Computer Emergency Response Team (CERT) defines a malicious insider as "A current of former employee, contractor, or business partner who has or had authorized access to a network and intentionally used that access in a way that negatively affect the confidentially, integrity, or availability of any information or information systems" [2]. Due to insider threats, cloud-based services are in serious risk of intellectual property theft, IT damage, and information leakage. Hence, security vulnerabilities emerging from insider threats should be addressed by policies, technical solutions, and proper detection methods.

2. SECURING SERVICES IN THE CLOUD

Service-Oriented Architecture (SOA) is an architectural pattern, while cloud computing is a set of enabling technologies as a potential target platform or technological approach for that architecture. By combining SOA and cloud computing, it becomes possible to reduce the time taken to implement technology, enhance business performance and expose the existing legacy application over the Internet. The role of SOA in cloud computing is important because a successful cloud solution requires an in-depth understanding of the architecture, the services offered and how to leverage them. Cloud computing becomes part of the architectural arsenal to create a successful SOA.

Cloud services benefit the business by taking the best practices and business process focus of SOA. These benefits apply to both cloud service providers and cloud service users. Cloud service providers need to architect solutions by using a service-oriented approach to deliver services with the expected levels of elasticity and scalability. Companies that architect and govern business processes with reusable service-oriented components can more easily identify which components can be successfully moved to public and private clouds. A Service-Oriented Architecture (SOA) is a software architecture for building business applications that implement business processes or services through a set of loosely coupled, black-box components orchestrated to deliver a well-defined level of service.

Service-oriented architectures are based on the idea of exposing software functionality as services to be used by independent parties. Their inherent independence of a specific platform and operating system make them perfectly suitable to connect service consumers and service providers over the Internet and provide a technical foundation for cloud computing [3]. The combination of SOA and cloud computing facilitating the provision of composed application and services that integrate and orchestrate services from different sources pose new challenges to security. Since services and applications are exposed to the Internet and are used in a global context, the management of user identities across organisational borders is a key element to perform access control and to prevent unauthorised access in a decentralised environment.

Open Identity Management Models support the sharing of identity information across several trust domains in a controlled manner. Clients can request identity information from the identity management systems and convey this information in an interoperable format to a requesting party. Besides identity provisioning, confidentiality and integrity of exchanged, stored, and processed information must be ensured. Several specifications emerged to protect information at different layers. For instance, a secure channel can be used to protect exchanged information, while signature and encryption mechanisms applied to a message can also protect stored and processed information. These security requirements are stated in security policies that configure the secure interaction of participants in a service-based system. Policies facilitate the negotiation of security requirements between services and service clients to enable interoperability at runtime. This enables a seamless usage of services in the cloud to build composed applications.

However, due to the complexity of the involved specifications, the variety of security mechanisms and the flexibility of service-based systems, such policies are hard to understand and even harder to codify. To overcome these limitations, we foster a model-driven approach that generates security configurations based on system design models annotated with security requirements. To implement the functional and security requirements specified at the modelling layer, our cloud platform has to ensure two aspects: The system with all involved services and web application components must be instantiated in a virtual machine according to the functional requirements and the services must be configured in compliance with the modelled security requirements. As illustrated in Figure 1, our approach consists of three layers. Functional and security requirements, expressed at the modelling layer, are translated to a platform independent model. This model constitutes the foundation to setup the virtual machine, application server, services and composed applications that are provided to the user [4].

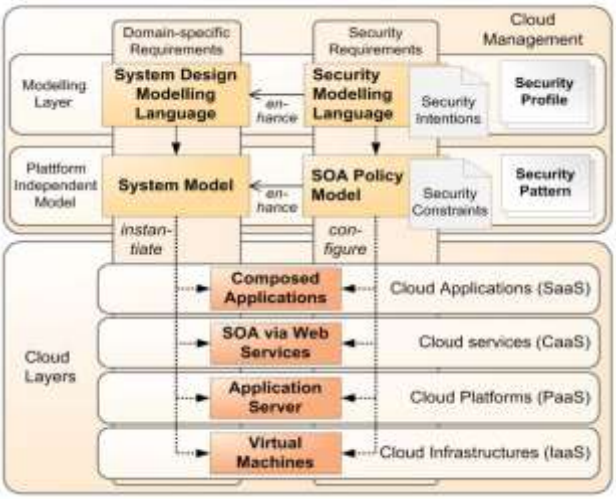

Fig -1: Model-driven Security in SOA

A Model-driven approach transforms security intentions to enforceable security policies. This transformation is based on a set of security configuration patterns that provide security expert knowledge to configure the system. Our model-driven approach requires an automated generation of enforceable security configurations based on the modelled security requirements [5].

3. DATA SECURITY AND PRIVACY IN CLOUD

Security and privacy concerns faced by the cloud consumers require them to evaluate the risk and its management in the cloud environment, then mitigating those risks. Of course, the most critical benefit offered by cloud computing is the reduction of business costs. Most businesses have well-established security objectives, strategies, and policies consistent with compliance requirements to protect their intellectual property, and their clients' data. Many security components come into play, but the most four critical components are shown in Figure 2. Data and transmission of data must take place through secured channels. Application and storage security both must be maintained by the cloud service provider.

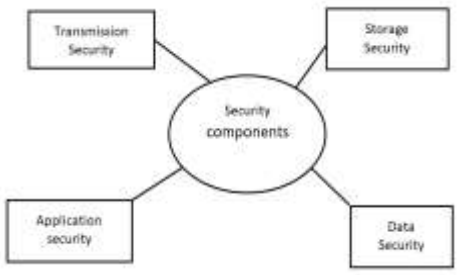

Fig -2: Security Components.

The storage data itself can be encrypted by many cloud service providers using server side encryption (SSE). This will encrypt the data on the cloud storage devices with strong cryptographic algorithms but the encryption keys stay with the cloud provider and are not under the control of the cloud customer. If nothing else is offered, the cloud server administrator should activate SSE per default for the cloud storage used. This encryption gives some basic protection against unauthorized access to the customer data but also means that the security of this solution depends on the cloud provider's ability to restrict access to the storage encryption keys even to other employees.

The cloud service provider should enforce the same or even higher levels of security controls as expected by the cloud customer or as best practice in the industry. There are logical risks of information disclosure or data integrity by having unsecure applications or permission handling functionalities. The application or underlying infrastructure could be open to exploits by hackers. The user permission and role model could be exploited as well by external hackers or internal employees that have too many access rights. In general, the same security measures need to be applied like in any IT system. The complexity arises from the cloud technology model that is based on virtualization and distributed responsibilities between the infrastructure layers. The cloud service provider must take care of physical and logical security that is in his sole responsibility. For example, the cloud service provider may offer encryption, but it is up to the customer to activate and use it. Clear responsibilities for network, operating system, and application security measures are key priorities to achieve such a secure cloud solution.

Cloud computing is yet to standardize the process of service metering. Therefore, service metering is not yet trustworthy to the cloud consumers. The process requires a systematic, verifiable, and reliable framework for cloud computing to be sustainable. Subsequently, the trust relationship of cloud service providers with customers and enterprises will be enhanced, resulting in a wider adoption of cloud-based solutions. Maintaining the privacy of users is of high concern for most organizations. Whether employees, customers, or patients, personally identifiable information is a high-valued target. Many cloud subscribers do not realize that when they contract a provider to perform a service, they are also agreeing to allow that provider to gather and share metadata and usage information about their environment. In some cases, providers even sell or share these data legally based on their privacy statements

The evolving nature of cloud computing technologies has resulted in nonstandard security implementations and practices. Moreover, the lack of governance for audits creates a challenging environment to verify if the cloud service providers have complied with the standards. As a result, cloud computing security may not yet be ready for audits [6]. Users depend on the service level agreement (SLA) and have to rely on the cloud service provider to keep up their end of the bargain. However, cloud services are best effort services and a service provider may not guarantee the

security standards. Therefore, as SLAs play a vital role in ensuring the security of the cloud-based services, governing bodies and security experts should be part of the SLAs and legal aspects, which is not yet seen to be in practice for cloud-based service models [7].

4. PROPOSED WORK

4.1 Operational Transformation

The edit conflict due to concurrent operations is one of the main challenges in collaborative editing systems. Without an efficient solution to edit conflicts, it may result in inconsistent text to different clients when collaborators concurrently edit the same document. In 2009, OT was adopted as a core technique behind the collaboration features in Apache Wave and Google Docs. In a collaborative editing cloud service, the cloud servers can be responsible for receiving and caching editing operations in its queue, imposing order on each editing operation, executing OT on concurrent operations based on the order iteratively, broadcasting these editing operations to other clients, and applying them in its local copy to maintain a latest version of the document. When receiving an operation $op_{r(c)}$ from the client, the cloud server executes operational transformation.

4.2. System Design and Security Analysis

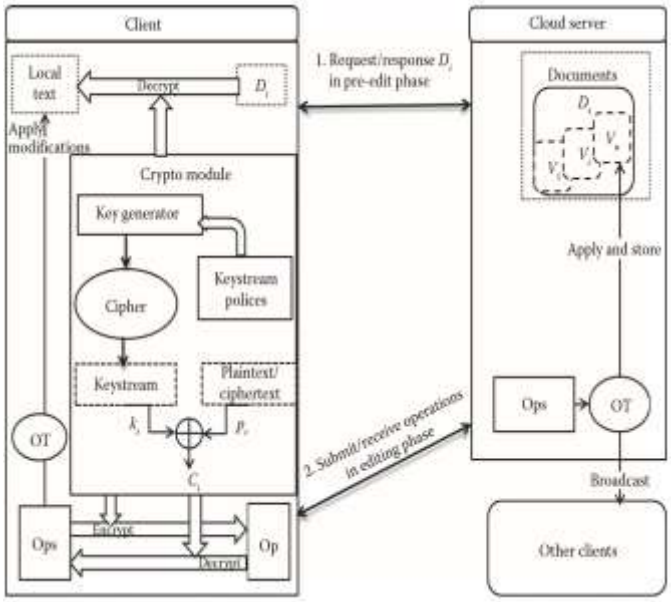

Fig -3: LightCore system model.

LightCore involves a group of collaborative users and a cloud server. Each client communicates with the server over the Internet, to send its operations and receive modifications from others in real time. For each document, the server maintains a history of versions. That is, it keeps receiving operations from users, and these modifications make the document shift from one version to another. When applying modifications on a version, the server may need OT to transform some operations. The server also keeps sending the current freshest version to users, that is, all transformed operations since the last version is sent. Because a user is still editing on the old version when the freshest one is being sent, the OT processing may also be required to update its view at the client side.

Figure 3 represents the LightCore system model, This model describes the cloud server is responsible for storing and maintaining the latest content, executing operations (update, delete, insert, etc.) against the content, resolving operational conflicts, and broadcasting the updates among multiple clients. The cloud server is considered to be honest but curious. In case of risking its reputation, the honest cloud server will timely and correctly disseminate modifications committed by all the authorized clients without maliciously attempting to add, drop, alter, or delay operation requests. However, motivated by economic benefits or curiosity, the cloud provider or its internal employees may spy or probe into the shared content, determine the document type by observing the format and layout, and discover the pivot part of the documents by analyzing the frequency and quantity of access. Additionally, we assume that the cloud servers will protect the content from unauthorized user's access and other traditional network attacks such as DoS attacks, and keep the availability of shared documents, for example, by redundancy.

LightCore system model is collaborative editing cloud solution for sensitive data against honest-but-curious servers. We adopt stream cipher or the CTR mode of block cipher to encrypt and decrypt the contents of the document within clients, while only the authorized users share the keys. In LightCore, all user data including all operations and every version of the documents are processed in the cloud. Attackers from inside or outside might attempt to alter or delete the user data, or disrupt the cloud services. However, for the reputation and benefits of the cloud service provider, the honest-but-curious cloud servers are supposed to preserve integrity, availability, and consistency for the data of users. The cloud service provider will deploy adequate protections to prevent such external attacks, including access control mechanisms to prevent malicious operations on a document by other unauthorized users. Preserving the confidentiality of users' documents is the main target of LightCore.

First, in our system, only the authorized users with the shared master key can read the texts of the documents. LightCore adopts stream cipher and the CTR mode of block cipher to encrypt data at the client side. In the editing phase, the input texts of each operation are encrypted before being sent to the cloud. Therefore, the input texts are transmitted in ciphertext and documents in the cloud are also stored in ciphertext. Second, the algorithms are assumed to be secure and the keys only appear on the clients. So, these keys could only be leaked by the collaborative users or the clients, who are also assumed to be trusted. Finally, data keys are generated in a random way by each user, and LightCore uses each byte of the keystreams generated by data keys only

once. Any text is encrypted by the keystreams generated specially for it. So, the curious servers cannot infer the contents by analyzing the difference in two decrypted texts.

In order to maintain the functionalities of the cloud servers, we only encrypt the input texts of each operation but not the position of the operation. The position of each operation and the length of the operated text are disclosed to the cloud servers, which may leak a certain of indirect sensitive information including the number of lines, the distribution of paragraphs, and other structure information. We assume these data can only be access by the authorized clients and the cloud servers, and they are not disclosed to external attackers by adopting the SSL protocol. In this case, the related data are limited to the cloud and the clients. Additionally, the attributes attached to the text segments, including font, color, author identity, keystream_info, might also be used to infer the underlying information of the documents. For example, a text segment with the bold attribute may disclose its importance; a text segment with list attribute may also leak some related information.

However, some of the attributes can be easily protected by encrypting them at the client in LightCore, because the cloud servers are not required to process all of them (e.g., font, size, and color). Therefore, encrypting these attributes will not impede the basic functionalities of the cloud servers. Anyway, attributes author and keystream_info cannot be encrypted, because these attributes related to the basic functionalities of the cloud servers. Another threat from the cloud is to infer sensitive data by collecting and analyzing data access patterns from careful observations on the inputs of clients. Even if all data are transmitted and stored in an encrypted format, traffic analysis techniques can reveal sensitive information about the documents.

Table -1: Performance of Concurrent Modifications from 20 Clients

	Queuing Time (ms)	Applying Time (ms)	Transmission Time (ms)	Decryption Time (RC4) (ms)	Total Time (ms)
Original System	0.06	5.93	23	-	1210
LightCore System	0.06	5.93	23	0.40	1239

In order to evaluate the time of these main procedures, we create an experiment where 20 collaborators from different clients quickly input texts in the same document concurrently. The time of transforming an operation called the queuing time, the time of applying an operation in its local copy called the applying time, and the transmission time of each operation are given in Table 1. In fact, the main difference lies in the added encryption/ decryption process; the other processes are not affected. The decryption time of less than 500 milliseconds has no influence on real time. We can see that the total time 1239 milliseconds of LightCore is only 29 milliseconds longer than that of the original system, which makes no difference to human perception.

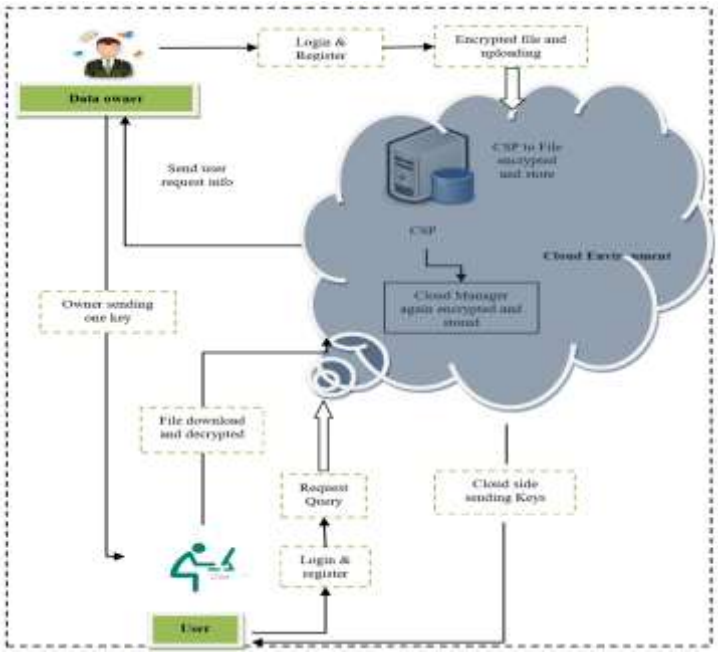

Fig -4: Multi-Layer Encryption Architecture

Multi layer Encryption Approach in Information Technology offers advanced and useful technique of cloud computing that is being intensively spreading among internet users. The present approach lacks safeguarding of uploaded data on the server also lacks proper tracking of abolished users thereby the user authentication process is hampered. A resolution can be that the user authentication process is performed twice. Firstly, authenticating the user by making use of password, thereafter using the interfacing technology in sending secret code to authentic users email. The data owner encrypts the data twice before uploading it cloud server. Later the cloud server and then the cloud manager again re-encrypt the data and finally stored in the cloud. The data to be uploaded on the cloud is granted additional security using Multi- layer encryption technique.

The paper projects performance of multi-layer encryption on the data to be uploaded on cloud server as data being more confidential and secure. The major concern of the paper is to perform the user authentication twice. Figure 4 suggests three-layers of encryption are to offer enhanced and higher safety of data and perform two types of login for user authentication. First, using the original user login name and password the validation is performed and secondly, to test whether user is authorized or not, a secret code is mailed to authentic users E-mail ID. On entering the correct secret code only the user is granted data access on cloud server. The purpose of authenticating the user twice is that only the authentic users can fetch the data from cloud server. The two

algorithms being used for achieving security are AES and RSA encryption techniques.

Table -2: Comparative of Accuracy

S.No.	Encryption Techniques	Security (%)	Time (ms)
1.	CP-ABE	88.2	3.58
2.	K-NN	92.1	2.71
3.	Multilayer Encryption Approach	96.6	0.96

Table 2 mentioned above demonstrates evaluation of Multi-layer encryption approach compared with K-NN, and CP- ABE. The suggested Multi-layer encryption technique proves to be effective providing enhanced performance in comparison with rest of the existing techniques.

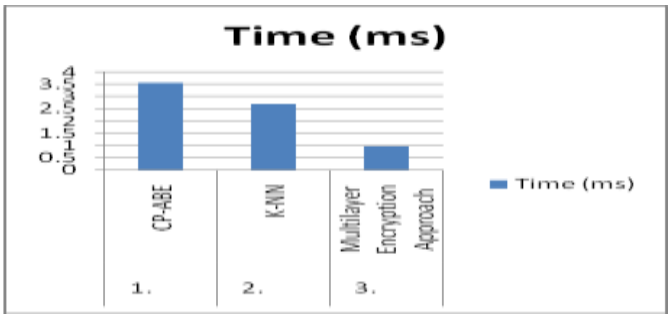

Chart -1: Comparison of Accuracy Analysis

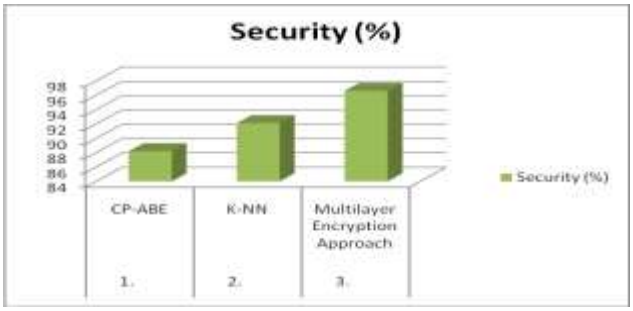

Chart -2: Comparison of time evolutions

Chart 1 and 2 mentioned above compares security based encryption models performing with Multi-layer encryption approach with K-NN, and CP- ABE. The suggested Multi-layer encryption technique is efficient offering greater performance compared with rest of the current techniques.

5. CONCLUSION

The proposed model-driven approach that transforms security intentions to enforceable security policies. This transformation is based on a set of security configuration patterns that provide security expert knowledge to configure the system. Our platform allows users to specify these security intentions in system models to enable a simple and easily comprehensible specification of security requirements. Experiments on the prototype system show that LightCore provides efficient online collaborative editing services for resource-limited clients. We adopt stream cipher or the CTR mode of block cipher to encrypt (and decrypt) the contents of the document within clients, while only the authorized users share the keys. This paper also presents a new strategy of Multi layer encryption oriented on the algorithm of AES and RSA which claims to offer security and privacy of entire public cloud content. Such data communication among the systems will lead to improvised security concerning the data that is shared over the cloud. This solution is helpful if the cloud customer wants full control not only over the encryption keys but also the key management and the encryption algorithms used.

REFERENCES

[1] Kincaid, J. (2010). Google confirms that it fired engineer for breaking internal privacy policies, available at http://techcrunch.com/2010/09/14/google-engineerspying-fired/.

[2] Silowash, G., Cappelli, D., Moore, A., Trzeciak, R., Shimeall, T. J., and Flynn, L. (2012). Common sense guide to mitigating insider threats, 4th edition, Tech. rep., DTIC Document.

[3] M. Jensen, J. Schwenk, N. Gruschka, and L. L. Iacono, "On technical security issues in cloud computing," Cloud Computing, IEEE International Conference on, vol. 0, pp. 109–116, 2009.

[4] Thalmann, S., Bachlechner, D., Demetz, L., and Maier, R. (2012). "Challenges in cross-organizational security management", in System Science (HICSS), in 45th Hawaii International Conference on (IEEE), pp. 5480–5489.

[5] M. Menzel, R. Warschofsky, and C. Meinel, "A Patterndriven Generation of Security Policies for Serviceoriented Architectures," in IEEE International Conference on Web Services (ICWS 2010), 2010.

[6] M. Menzel and C. Meinel, "SecureSOA - Modelling Security Requirements for Service-oriented Architectures," in IEEE International Conference on Services Computing (SCC 2010), 2010.

[7] Morin, J.-H., Aubert, J., and Gateau, B. (2012). "Towards cloud computing SLA risk management: Issues and challenges", in System Science (HICSS), 2012 45th Hawaii International Conference on (IEEE), pp. 5509–5514.

BIOGRAPHIES

Mr. Y Kiran Kumar, M.C.A. He received his Master of Computer Applications from Sri Venkatesra University, Tirupati. He is Pursuing Ph.D from Bharathiar University, Coimbatore. He is having more than 11 years of teaching experience, currently he is working as a Assistant Professor in the department of M.C.A in Sree Vidyanikethan Engineering College, Affiliated by JNTUA, Ananthapuramu, India. His areas of research interests include Web Technologies, Information Security, Service Oriented Architecture and Cloud Computing.

Dr. R. Mahammad Shafi, M.C.A, M.Tech, Ph.D. He received his Ph.D from University of Allahabad, Allahabad. He is having more than 20 years of teaching experience. His areas of research interests include Software Engineering, Software Testing and Quality Assurance. He has published papers in refereed journals and conference proceedings in these areas. He has been involved in conferences and workshops as a Committee member, organizer and Session Chair. His areas of research interests include Software Engineering, Software Testing and Quality Assurance.