2010 34th Annual IEEE Computer Software and Applications Conference Workshops

SecureCloud: Towards a Comprehensive Security Framework for Cloud Computing Environments

Hassan Takabi
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA, USA
hatakabi@sis.pitt.edu

James B. D. Joshi
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA, USA
jjoshi@sis.pitt.edu

Gail-Joon Ahn
School of Computing, Informatics,
and Decisions Systems Engineering
Arizona State University
Tempe, AZ, USA
gahn@asu.edu

***Abstract*—Cloud computing has recently gained tremendous momentum but still is in its infancy. It has the potential for significant cost reduction and the increased operating efficiencies in computing. Although security issues are delaying its fast adoption, cloud computing is an unstoppable force and we need to provide security mechanisms to ensure its secure adoption. In this paper, we propose a comprehensive security framework for cloud computing environments. We also discuss challenges, existing solutions, approaches, and future work needed to provide a trustworthy cloud computing environment.**

I. Introduction

Cloud computing has recently raised an intensive interest within both academic and industry communities. It is still an evolving paradigm that incorporates the evolutionary development of many existing computing technologies such as distributed services, applications, information and infrastructure consisting of pools of computers, networks, information and storage resources [2]. It has shown tremendous potential to enhance collaboration, agility, scale, availability–although its definitions, issues, underlying technologies, risks, and values need to be refined. These definitions, attributes, and characteristics have been evolving and will change over time. The US National Institute of Standards and Technology (NIST) defines cloud as follows:"Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three delivery models, and four deployment models." [1].

Cloud computing has become a very attractive paradigm because of its perceived economic and operational benefits. Despite the enormous opportunity and value that the cloud presents for organizations, several surveys of potential cloud adopters indicate that security and privacy are the number one concern delaying its adoption and it will continue to keep some companies out of cloud computing [3]. For example, on March 30, 2010, Yale University placed a migration to Google Apps for its email services on hold over privacy and security concerns [21]. However, cloud computing appears to be an unstoppable force because of its potential benefits. Hence, understanding the security and privacy risks in cloud computing and efficient and developing effective solutions are critical to the success of this new computing paradigm. When we move our information into the cloud, we may lose control of it. The cloud gives us access to the data, but the challenge is to ensure that only authorized entities have access to the data. It is crucial to understand how we can protect our data and resources from a security breach in the cloud that provides shared platforms and services. It is critical to have appropriate mechanisms to prevent cloud providers from using customers' data in a way that has not been agreed upon in the past.

In this paper, we propose a comprehensive security framework for cloud computing environments. We present the security framework and discuss existing solutions, some approaches to deal with security challenges. The framework consists of different modules to handle security, and trust issues of key components of cloud computing environments. These modules deal with issues such as identity management, access control, policy integration among multiple clouds, trust management between different clouds and between a cloud and its users, secure service composition and integration, and semantic heterogeneity among policies from different clouds.

The remainder of this paper is organized as follows: Section 2 presents an overview of unique features of the cloud with their security implications. Section 3 presents the security framework followed by components of the framework and some approaches in Section 4. Finally, Section 5 concludes the paper and discusses the future work.

II. Overview of cloud and its features

The five key characteristics of cloud computing include *on-demand self-service*, *ubiquitous network access*, *location independent resource pooling*, *rapid elasticity*, and *measured service*. Rapid elasticity allows resources provisioned to be quickly scaled up and down. Measured services indicate that a cloud service provider controls and optimizes the use of computing resources through automated resource allocation, load balancing and metering tools [1].

The three cloud delivery models are as follows: *Software as a Service* (SaaS), *Platform as a Service* (PaaS), and *Infrastructure as a Service* (IaaS). In IaaS, the cloud provider provides a set of virtualized infrastructural components such

978-0-7695-4105-1/10 $26.00 © 2010 IEEE
DOI 10.1109/COMPSACW.2010.74

as virtual machines and storage on which the customers can build and run applications. Issues such as trusting the virtual machine images, hardening hosts, and securing inter-host communication are critical areas in IaaS. PaaS enables the programming environment to access and utilize additional application building blocks. Such a programming environment has a visible impact on the application architecture. One such impact could be the constraints on the services that the application can request from an OS. In SaaS, the cloud providers provision application software as on-demand-services. As clients acquire software components from potentially different providers, securely composing them and ensuring that information handled by these composed services are well protected become crucial issues.

The cloud deployment models can be categorized as *public cloud*, *private cloud*, *community cloud*, and *hybrid cloud* composed of multiple clouds. Public clouds are publicly available and can serve multiple tenants, while private cloud is typically a tailored environment with dedicated virtualized resources for a particular organization. Similarly, community cloud is tailored for a particular group of customers.

A. Security Implications of Cloud Features

The architectural features of the cloud allow users to achieve better operating costs and be very agile by facilitating fast acquisition of services and infrastructural resources as and when needed. However, these unique features also give rise to various security concerns. Table I summarizes these unique features with corresponding security implications [20].

B. Cloud Example

Here, we provide an example to show what security and privacy issues arise when an organization migrates to the cloud. Suppose an organization wants to adopt cloud computing. It uses Amazon S3 and FlexiScale, which are examples of IaaS for storage and maintaining virtual servers respectively. As instances of PaaS, Google App Engine and LoadStorm are used for running web applications and testing their performance respectively. It also uses the Zoho, Zuora, Workday, Clickability, Salesforce, and DocLanding that are instances of SaaS for different purposes such as email, billing, content management, human resource management, etc. In the following, we provide a brief description of what each service provides:

- Amazon S3 provides "a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web" (http://aws.amazon.com/s3).
- FlexiScale provides virtual dedicated servers to its customers (http://www.flexiant.com/products/flexiscale).
- Google App Engine provides web applications on Google's infrastructure. When using Google App Engine, there are no servers to maintain; the organization just uploads its application to serve users (http://code.google.com/appengine).
- LoadStorm enables web developers to improve the performance of their web applications. One can create his/her own test plans, and generate concurrent users of http traffic in realistic scenarios (http://loadstorm.com).
- Zoho offers a set of web applications geared towards increasing productivity and offering convenient collaboration (http://www.zoho.com).
- Zuora offers online recurring billing and payment solutions for SaaS and subscription businesses (http://www.zuora.com).
- Workday provides human resource management, financial management, and payroll, and delivers the solutions on an SaaS model (http://www.workday.com).
- Clickability offers on demand web content management by combining the benefits of SaaS with IaaS (http://www.clickability.com).
- Salesforce provide customer relationship management (CRM) service to its customers (http://www.salesforce.com).
- DocLanding offers on-demand web-based document management service (http://www.doclanding.com).

III. Security Challenges in Cloud

The cloud computing can be deemed as an instance of the multi-domain environment where each domain employs different security and trust requirements and potentially employs various mechanisms and semantics. Such domains could represent individually enabled services or application components. Service-oriented-architecture (SOA) is naturally relevant technology to facilitate such multi-domain formation through service composition [3]. The existing research on multi-domain policy integration and the secure service composition can be leveraged to build a comprehensive security framework in the cloud computing environment. Here, we discuss the key security challenges that cloud computing raises.

A. Authentication and Identity Management

By using cloud services users can easily access their personal information and it is also available to various services across the Internet. We need to have an identity management (IDM) mechanism for authenticating users and services based on their credentials and/or profile/characteristics [11]. One key issue concerning IDM in the cloud is the interoperability issues that could result from using different identity tokens and different identity negotiation protocols. An IDM system should be able to accommodate privacy concerns related to protection of private and sensitive information associated with users and processes. How the multitenant cloud environment could affect the privacy of identity information has not been yet well understood. When a user interacts with a front end service, this service may need to ensure that his/her identity is protected from other services that it interacts with [11], [12].

B. Access Control

Heterogeneity and diversity of services, and the domains' diverse access requirements in cloud computing environments

TABLE I
SECURITY IMPLICATIONS OF CLOUD FEATURES

Feature	Security Implication
Outsourcing	Users may lose control of their data. Appropriate mechanisms needed to prevent cloud providers from using customers' data in a way that has not been agreed upon in the past.
Extensibility and Shared Responsibility	There is a tradeoff between extensibility and security responsibility for customers in different delivery models.
Virtualization	There needs to be mechanisms to ensure strong isolation, mediated sharing and communications between virtual machines. This could be done using a flexible access control system to enforce access policies that govern the control and sharing capabilities of VMs within a cloud host.
Multi-tenancy	Issues like access policies, application deployment, and data access and protection should be taken into account to provide a secure multi-tenant environment.
Service Level Agreement	The main goal is to build a new layer to create a negotiation mechanism for the contract between providers and consumers of services as well as the monitoring of its fulfillment at run-time.
Heterogeneity	Different cloud providers may have different approaches to provide security and privacy mechanisms, thus generating integration challenges.

would require fine-grained access control policies. In particular, access control services should be flexible enough to capture dynamic, context or attribute/credential based access requirements, and facilitate enforcement of the principle of least privilege. Such access control services may need to integrate privacy protection requirements derived from complex rules. It is important that the access control system employed in clouds is easily managed and its privilege distribution is administered efficiently. It is also important to ensure that the cloud delivery models provide generic access control interfaces for proper interoperability, which demands for a policy neutral access control specification and enforcement framework that can be used to address cross-domain access control issues [13]. Also, the access control models should be able to capture relevant aspects of SLA agreements.

C. Policy Integration

In our example, Amazon, Google, and FlexiScale and other providers have their own policies which need to be integrated in a secure manner when the customer uses them together. It has been shown that even when an individual provider's policies are verified to be correct, security violation can easily occur as they are integrated [14]. The policy integration task in the cloud should be able to address challenges such as semantic heterogeneity, secure interoperability, and policy evolution management. It is important to have secure interoperation mechanisms to ensuring that no security breaches are created during the interoperation. In addition, policy engineering mechanisms are needed to integrate access policies of different cloud service providers and define global access policies to accommodate all collaborators' requirements. Furthermore, semantic heterogeneity exists among policies from different service providers that need to be taken into account. Amazon, Google, LoadStorm and other providers may have their own approaches and there might be semantic conflicts and/or inconsistencies among their policies. There is a need to automatically detect these possible conflicts and resolve them.

D. Service Management

In the cloud computing environment, cloud service providers collaborate to provide desirable newly composed services that meet customers' needs. In our example, there could be a service integrator that composes Zoho, Workday, and Zuora to form a new composed service and provides a packaged service to customers. Although many cloud service providers provide their services with Web Services description language (WSDL), the traditional WSDL cannot fully meet the requirements of cloud computing services description. In clouds, issues like QoS, service price, and SLAs are critical in service search and service composition. These issues need to be accommodated in describing services with unified standards to introduce their features, to find best interoperable services, to integrate them without violating the service owner's policies, and to ensure SLAs are satisfied.

E. Trust Management

Suppose in the example, the customer designs and runs its own web applications on Google's infrastructure but it needs to test performance of the designed web applications using LoadStorm. Does the customer trust LoadStorm? Do Google and LoadStorm trust each other? How can they negotiate the trust? Is the trust static/dynamic? What are the requirements to manage trust? In cloud computing environments, the interactions between different service domains driven by service-requirements can be expected to be very dynamic/transient and intensive. Thus, a trust management framework should be developed to efficiently capture a generic set of parameters required for establishing trust and to manage evolving trust and interaction/sharing requirements. Furthermore, the customers' behavior can evolve rapidly, thereby affecting established trust values. Efficient techniques are needed to manage evolving trust. This suggests a need for a trust management approach to support the establishment, negotiation and maintenance of trust to adaptively support policy integration [14], [16]. There exist some critical questions that need to be answered: How do we establish trust and determine access mapping to satisfy inter-domain access requirements? How do we manage and maintain dynamically changing trust values and adapt the access requirements as trust evolves?

IV. SECURITY FRAMEWORK FOR THE CLOUD

In this section, we provide an overview of our proposed security framework for the cloud computing environment and then articulate some approaches to address its components.

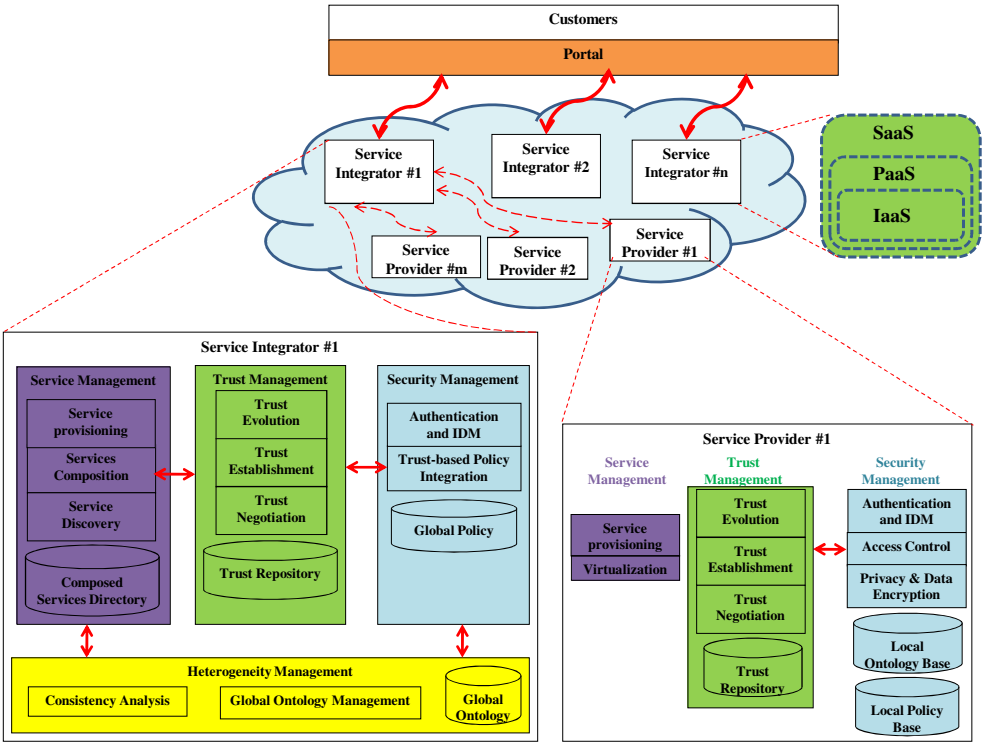

Fig. 1. Security Framework For Cloud Computing Environments

The overall security framework and key components of the cloud computing environment are depicted in Figure 1. The *Services Integrator* facilitates collaboration among different service providers by composing new desirable services. Each service integrator has components that are responsible for establishment and maintenance of trust between the local provider domains and between the providers and the users, provisioning desirable services and generating global policies. The service integrators first discover services from different service providers or other service integrators, carry out negotiations, integrate the services to form groups of collaborating services and provide them to users.

The *security management* component provides the security and privacy specification and enforcement functionality. The *authentication and identity management module* is responsible for authenticating users and services based on credentials and characteristics. In service provider, the *access control module* employs the access policies while the *privacy and data encryption module* is responsible for privacy needs and encryption of outsourced data. In the service integrator, the *trust-based policy integration (TPI) module* is the key component that administers trust and facilitates trust-based policy integration among different services from different service providers. The *service management* component is responsible for secure service discovery, composition and provisioning. The service provider uses *virtualization* in order to offer services to users more efficiently. The *service discovery module* is responsible for finding different services that the provider domains or other service integrators offer. After discovering services, the service integrator needs to negotiate with the provider domains and compose new desirable collaborating services for users using the *service composition module*. The collaborating services come from different domains and the service integrator needs to consider trust between the collaborating provider domains when composing new services. The *service provisioning module* provides services for users based on bidirectional trust between the service integrator and its users. The *trust management* is responsible for negotiation, establishment, and evolution of trust. The *global ontology management module* is responsible for providing global ontology and supporting semantic heterogeneity concerns related to policies. The *consistency analysis module* is used to check the correctness of the integrated policies.

A. Access Control Module

This module is responsible for supporting providers' access control needs. Based on the requirements, various access control models can be used. Role Based Access Control (RBAC) has been widely accepted as the most promising access control model because of its simplicity, flexibility in capturing dynamic requirements, and support for the principle of least privilege and efficient privilege management [4], [13]. Furthermore, RBAC is policy neutral, can capture a wide variety of policy requirements, and is best suited for policy integration needs discussed earlier. RBAC can also be used for usage control purpose which generalizes access control

to integrate obligations and conditions into authorizations. Obligations are defined as requirements that the subjects have to fulfill for access requests. Conditions are environmental requirements independent from subject and object that have to be satisfied for the access request. Due to the highly dynamic nature of the cloud, obligations and conditions are crucial decision factors for richer and finer controls on usage of resources provided by the cloud. Recent RBAC extensions such as credential-based RBAC [7], Generalized Temporal RBAC (GTRBAC) [4], and location based RBAC models provide necessary modeling constructs and capabilities to capture context based fine-grained access control requirements. In clouds, users are usually not known a priori to the service providers so it is difficult to assign users directly to roles in access control policies - use of credential/attribute based policies may enhance this capability. However, little work exist in employing RBAC and extensions within intensely service-oriented environments such as clouds.

B. Policy Integration Module

Existing work on multi-domain access control policies have addressed the issue of (i) integrating policies to ensure secure interoperation and (ii) policy engineering mechanisms to integrate access policies of different policy domains and define global access policies [14], [15]. Some approaches include policy algebra that can facilitate specification of various combinations of policies from different policy domains. Secure interoperation can be achieved in a centralized or decentralized fashion [14]. In a centralized approach, a global policy is created to mediate all accesses and is appropriate for cloud application that is statically composed of various services with different requirements. In a more dynamic environment, the domains are transient and may need to interact for a very specific purpose. More decentralized approaches are needed in such cases. Specification frameworks are needed to ensure that the cross domain accesses are properly specified, verified and enforced. SAML, XACML, and WS standards are viable solutions towards this needs [14]. However, support for fine-grained RBAC capabilities may be limited as indicated by RBAC specific multi-domain policy specification and enforcement frameworks in XRBAC [13]. Policy engineering mechanisms are crucial to define global policies to accommodate all collaborators' requirements. Emerging role mining techniques can be useful to support this [5]. In the cloud, users acquire different roles from different domains based on services they need. To define global policies, we can utilize these RBAC systems' configurations from different domains to define global roles and policies. This process also needs to address policy evolution or changing requirements. One possible approach is StateMiner [5] that presents a heuristic role mining solution and could be adopted in clouds for policy engineering.

C. Service Management Module

An automatic and systematic service provisioning and composition approach that considers security and privacy issues needs to be developed. Researchers have developed ways to configure and map the Open Services Gateway Initiative (OSGi) (http://www.osgi.org) authorization mechanism to RBAC [17]. Declarative OWL-based language can be used to provide a service definition manifest including a list of distinct component types that make up the service, the functional requirements, component grouping and topology instructions, etc. OSGi can be adopted to develop an agent-based collaboration system for automatic service provisioning. A key aspect of collaboration systems is the group situation that changes dynamically and governs the requirements of the collaboration. Individual agent context is important to characterize the situation of the agent, and the overall cooperative behaviors are driven by the group context because of relationships and interactions among agents. The Group Situation based-RBAC model [6] could be extended to address provisioning services based on group situation. The model emphasizes a capability based agent to facilitate role mapping and group situation driven permission assignment to cope with dynamic access policies that evolve continuously.

D. Trust Management Module

In the cloud, there is a challenging need of integrating requirements-driven trust negotiation techniques with fine-grained access control mechanisms. Due to the cloud's nature that is service oriented, the trust level should also be integrated with the service. The idea is that the more services a cloud service provider provides, the higher trust level needs to be established. Another problem is that we need to establish bi-direction trust in the cloud. That is, the users should have some level of trust on the providers to choose their services from, and the providers also need to have some level of trust on the users to release their services to. One possible approach is to develop a trust management approach that includes a generic set of trust negotiation parameters, is integrated with service, and is bi-directional. As the service composition dynamics in the cloud are very complex, trust as well as access control frameworks should include delegation primitives [16]. Existing work related to access control delegation, including role-based delegation, has been focused on issues related to delegation of privileges among subjects and various levels of controls with regard to privilege propagation and revocation. Efficient cryptographic mechanisms for trust delegation involve complex trust chain verification and revocation issues raising significant key management issues with regard to its efficiency [15].

E. Heterogeneity Management Module

It is necessary to address semantic heterogeneity among different service providers' policies since they may have different approaches to provide security mechanisms [3], [15], [14]. Little attention has been given to detection of semantic conflicts among different service providers' policies. While XML has been adopted as the preferred language for information sharing it has been found inadequate for describing information semantics [9], [10]. RDF, on the other hand,

provides a facility for describing semantics by supporting element attributes and properties description [8]. Although semantics can be captured using RDF, representing relations between the various concepts is essential for facilitating semantic integration of policy information within interacting domains. Ontology-based approach is the most promising method to address the semantic heterogeneity issue [19]. To support the development of ontologies, both XML-Schema and RDF-Schema can be used to accommodate the domain-specific concepts. An OWL based solution can be developed to support semantic heterogeneity across multiple providers in the cloud. In developing this solution, we can adopt a system-driven policy framework to facilitate the management of security policies in heterogeneous environments and a policy enforcement architecture [18], [19].

F. Authentication and Identity Management Module

In the cloud, an appropriate user-centric IDM is essential to provide a flexible, scalable IDM service. User-centric IDM has recently received significant attention for handling private and critical identity attributes [12]. In this approach, an identity has identifiers or attributes that identify and define the user. The notable idea of user-centric approach allows users to control their own digital identities and also takes away the complexity of IDM from the enterprises, therefore allowing them to focus on their own functions. User-centric IDM also implies that the system properly maintains the semantics of the context of identity information for users, and sometimes constrains and relaxes them in order to find the best way to respond to a given user request in a given situation. Other federated IDM solutions would also benefit heterogeneous cloud environments [11], [12]. It is important to ensure that if needed, IDM services in the cloud can be integrated with an enterprise's existing IDM framework [2], [3]. In some cases it is important to have privacy-preserving protocols to verify various identity attributes. The zero-knowledge proof based techniques can be used for this purpose [11]. Existing techniques for use of pseudonyms and accommodating multiple identities to protect users' privacy can help build a desired user-centric federated IDM for clouds. IDM solutions can be further extended with delegation capabilities to address identification and authentication issues in complex service composition environments.

V. Conclusion

Cloud computing is still in its infancy and although security issues are delaying its adoption, it is growing and we need to provide security mechanisms to ensure that cloud computing benefits are fully realized. In this paper, we have presented a comprehensive security framework for cloud computing environments. We have described its components, discussed existing solutions and identified possible approaches to deal with different security issues related to the cloud.

Acknowledgment

The work of Gail-J. Ahn was partially supported by the grants from National Science Foundation (NSF-IIS-0900970 and NSF-CNS-0831360). James Joshi is also partially supported by the grants from National Science Foundation (NSF-IIS-0545912).

References

[1] http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc
[2] Cloud Security Alliance Report,"Security Guidance for Critical Areas of Focus in Cloud Computing V2.1" (http://cloudsecurityalliance.org/)
[3] Daniele Catteddu, Giles Hogben, (ENISA report) "Cloud Computing: Benefits, risks and recommendations for information security," (http://www.enisa.europa.eu/act/rm/files/deliverables/cloud-computing-risk-assessment/at_download/fullReport)
[4] James B. D. Joshi, Elisa Bertino, Usman Latif, and Arif Ghafoor, "A Generalized Temporal Role-Based Access Control Model", IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 1, 2005.
[5] Hassan Takabi and James B. D. Joshi, "StateMiner: An Efficient Similarity-Based Approach for Optimal Mining of Role Hierarchy", In Proc. of the 15th ACM symposium on access control models and technologies (SACMAT10), USA, ACM Press, 2010.
[6] Minsoo Kim, James B.D. Joshi, Minkoo Kim, "Access Control for Cooperation Systems based on Group Situation," In Proc. of the 4th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom2008), USA, 2008.
[7] Sudip Chakraborty and Indrajit Ray, "TrustBAC: integrating trust relationships into the RBAC model for access control in open systems", In Proc. of the 11th ACM symposium on access control models and technologies (SACMAT06), pages 49-58, ACM Press, 2006.
[8] N. B. Kodali, C. Farkas, and D. Wijesekera, "Specifying Multimedia Access Control using RDF," Journal of Computer Systems, Science and Engineering, vol. 19, 2004.
[9] R. Bhatti, J. B. D. Joshi, E. Bertino, and A. Ghafoor, "Access Control in Dynamic XML-based Web-Services with X-RBAC," The First International Conference in Web Services, USA, 2003.
[10] R. Bhatti, J. B. D. Joshi, E. Bertino, and A. Ghafoor, "X-GTRBAC An XML-based Policy Specification Framework and Architecture for Enterprise-Wide Access Control," ACM Transactions on Information and System Security Vol. 8, No. 2, 2005.
[11] Elisa Bertino, Federica Paci, Rodolfo Ferrini, "Privacy-preserving Digital Identity Management for Cloud Computing," IEEE Computer Society Data Engineering Bulletin, pages 1-4, 2009.
[12] Moonam Ko, Gail-joon Ahn, Mohamed Shehab "Privacy enhanced User-Centric Identity Management," In Proceedings of IEEE International Conference on Communications, Dresden, Germany, June 14-18, 2009.
[13] James Joshi, Rafae Bhatti, Elisa Bertino, Arif Ghafoor, "Access-Control Language for Multidomain Environments," IEEE Internet Computing, Vol. 8, No. 6, 2004.
[14] Yue Zhang, James Joshi, "Access Control and Trust Management for Emerging Multidomain Environments," in Annals of Emerging Research in Information Assurance, Security and Privacy Services, Editors: S. Upadhyaya, R. O. Rao 2009.
[15] Matt Blaze, Sampath Kannan, Insup Lee, Oleg Sokolsky, Jonathan M. Smith, Angelos D. Keromytis, and Wenke Lee, "Dynamic Trust Management," IEEE Computer, pages 44-51, 2009.
[16] Dongwan Shin and Gail-J. Ahn, "Role-based Privilege and Trust Management," Computer Systems Science and Engineering Journal, Vol. 20, No. 6, CRL Publishing, 2005.
[17] Gail-Joon Ahn, Hongxin Hu and Jing Jin, "Security-enhanced OSGi Service Environments," IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, Vol. 39, No. 5, 2009.
[18] Lawrence Teo and Gail-Joon Ahn, "Managing Heterogeneous Network Environments Using an Extensible Policy Framework," In Proc. of the ASIAN ACM Symposium on Information, Computer and Communications Security (ASIACCS07), Singapore, ACM Press, 2007.
[19] Hassan Takabi, Minsoo Kim, James B. D. Joshi, and Michael B. Spring, "An architecture for specification and enforcement of temporal access control constraints using OWL," In Proc. of the 2009 ACM workshop on Secure web services (SWS'09), ACM Press, 2009.
[20] Hassan Takabi, James B. D. Joshi, and Gail-Joon Ahn, "Security and Privacy Challenges in Cloud Computing Environments," Technical Report, 2010.
[21] http://www.yaledailynews.com/news/university-news/2010/03/30/its-delays-switch-gmail-community-input/