# Information Flow Control in Cloud Computing

Ruoyu Wu[1], Gail-Joon Ahn[1], Hongxin Hu[1], Mukesh Singhal[2]
[1]Laboratory of Security Engineering for Future Computing (SEFCOM), Arizona State University, Tempe, AZ 85287, USA
[2]Department of Computer Science, University of Kentucky, Lexington, KY 40506, USA
Email: {ruoyuwu,gahn,hxhu}@asu.edu, singhal@cs.uky.edu

***Abstract*—Cloud computing is an emerging computing paradigm where computing resources are provided as services over Internet while residing in a large data center. Even though it enables us to dynamically provide servers with the ability to address a wide range of needs, this paradigm brings forth many new challenges for the data security and access control as users outsource their sensitive data to clouds, which are beyond the same trusted domain as data owners. A fundamental problem is the existence of insecure information flows due to the fact that a service provider can access multiple virtual machines in clouds. Sensitive information may be leaked to unauthorized customers and such critical information flows could raise *conflict-of-interest* issues in cloud computing.**

**In this paper, we propose an approach to enforce the information flow policies at Infrastructure-as-a-Service (IaaS) layer in a cloud computing environment. Especially, we adopt Chinese Wall policies to address the problems of insecure information flow. We implement a proof-of-concept prototype system based on Eucalyptus open source packages to show the feasibility of our approach. This system facilitates the cloud management modules to resolve the conflict-of-interest issues for service providers in clouds.**

## I. Introduction

Although cloud computing is based on a collection of many existing and few new concepts in several research areas like service-oriented-architecture (SOA) [11], distributed and grid computing [12], [13] as well as virtualization [7], [23], it has become a promising computing paradigm drawing extensive attention from both academia and industry. This paradigm shifts the location of computing infrastructure to the network as service associated with the management of hardware and software resources. It has shown tremendous potential to enhance collaboration, scale, agility and availability.

Along with this new paradigm, various cloud service delivery models are developed, which can be divided into three layers [22] depending on the type of resources provided by the cloud. The bottom-most layer provides fundamental computing resources such as processing, storage, networks and is, henceforth, denoted as IaaS. A consumer is able to deploy and run arbitrary softwares, which include operating systems and applications. Amazon's EC2 and S3 [1] are prominent examples for IaaS in cloud computing. On the top of IaaS, more platform-oriented services allow the usage of hosting environments tailored to a specific need. Google App Engine [2] is an example for a Platform-as-a-Service (PaaS) which enables to deploy and dynamically scale Python and Java based Web applications. The top-most layer provides its users with ready to use applications running on a cloud infrastructure, also known as Software-as-a-Service (SaaS). Users can access those applications through a thin client interface such as a Web browser. Salesforce Customer Relationships Management [4] is an example of SaaS.

All of those services provide users with scalable resources in the pay-as-you-go fashion at relatively low costs. For example, Amazon's EC2 sells 1.0-GHz x86 ISA 'slices' for $0.10 per hour, and a new 'slice', or instance, can be added in 2 to 5 minutes. Amazon's S3 charges $0.12 to $0.15 per gigabyte-month, with additional bandwidth charges of $0.10 to $0.15 per gigabyte to move data into and out of Amazon Web Services over the Internet [5]. Comparing with building and managing their own infrastructures, users are able to save their investments significantly by migrating businesses to a cloud. With the increasing development of cost-effective cloud computing technologies, it is not hard to imagine that more and more businesses will be adopting cloud computing in the near future.

As promising as it is, cloud computing is also facing many security challenges [21] including authentication and identity management, access control, policy integration and so on. If not properly resolved, those challenges may hinder cloud computing's fast growth. Our work focuses on access control issues in cloud computing environments that would raise great concerns from customers, which can be of individuals, organizations, or enterprises when they outsource sensitive data to clouds. These concerns are traceable to the fact that cloud infrastructures are usually operated by commercial service providers that are outside of the trusted domain of the users, even in another country with a different regulatory environment. Insecure information flows [17] exist in clouds at a very high rate since a service provider can access multiple cloud virtual machines where various customers' data are stored. This can raise conflict-of-interest issues when the service provider discloses sensitive information of a customer to other competing customers for commercial profits, which can cause tremendous loss to a customer. This problem is more obvious when consulting services are migrated into clouds. It is natural that consultants have to deal with confidential information stored in clouds for their customers.

Consider a scenario shown in Figure 1 where a service provider provides business consulting services [3] using cloud infrastructure. His customers consist of banks including Bank of America (BoA), Chase and HSBC, and airline companies including United Airlines (UA) and Delta Air Lines (Delta). All of his customers need to outsource their consulting related data to cloud virtual machines running on the cloud

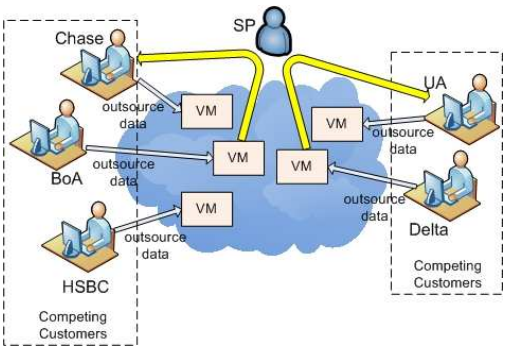



Fig. 1: Insecure Information Flows in Clouds

infrastructure. Suppose UA is trying to purchase airplanes to open up new routes and needs investments from banks. All the three banks are willing and competing to provide the investments to UA because of their business and financial interests. Since the consultant can access all the VMs in clouds, it is very likely the consultant will help one bank gain the contract with UA by leaking biding information of the other banks because of personal gains. In that case, the other banks will have tremendous commercial loss. Both UA and Delta also have sensitive information regarding plans, status and standing stored in clouds, each of whom wants to inquire through the consultant for competition. The consultant may also inadvertently disclose one's sensitive information to the other when serving both UA and Delta at the same time. The yellow arrows in the Figure 1 show two insecure information flows in clouds. The service provider discloses the sensitive information of BoA to Chase and the sensitive information of Delta to UA. This scenario demonstrates the possible existence of information flow problem in cloud computing which in turns raises conflict-of-interest issues and a critical need to investigate corresponding countermeasures.

In this paper, we propose an approach to enforce the information flow policies at IaaS layer in a cloud computing environment. Especially, we adopt Chinese Wall policies [9] to address the problems of insecure information flow. We implement a proof-of-concept prototype system based on Eucalyptus open source packages [19], [20] to show the feasibility of our approach. This system facilitates the cloud management modules to resolve the conflict-of-interest issues for service providers in clouds.

The rest of this paper is organized as follows. We give an overview of the information flow policies focusing on Chinese Wall security policy in Section II. In Section III, we present an approach to enforce Chinese Wall security policy in cloud computing at IaaS layer, which can be used to eliminate insecure information flow problem in clouds. Section IV describes the system design of our prototype cloud management system. Section V presents the implementation details followed by the related work in Section VI. Finally, in Section VII we conclude the paper with a summary of our results and a discussion of issues that remain to be addressed.

## II. Overview of the Chinese Wall Security Policy

Security policy research was derived from the formal definition of military security policy, succeeded by the Bell-LaPadula [8], [16]. In 1987, Clark and Wilson drew much attention to the importance of commercial security policy models in their seminal paper [10]. They claimed that the needs of the commercial community are just as important as the needs of the military community. Furthermore they emphasized that the problems of the commercial community are diverse and therefore require their own security policy models. All of the above-mentioned policy models were designed to operate in a well-defined environment, ranging from a strict military environment to a commercial environment.

Brewer and Nash introduced the Chinese Wall Security Policy [9] that makes use of subjects and objects to prevent information flows which cause conflict-of-interests for an individual consultant. All company information is stored in a hierarchical file system shown in Figure 2 which consists of three levels:

1) The lowest level consists of individual objects of information, each being associated with a single company.
2) The intermediate level consists of company datasets which group all objects concerning the same company together.
3) The highest level consists of conflict of interest classes which group all company datasets whose companies are in the competition together.

Each individual object is associated with the name of the company dataset to which it belongs. Similarly, each company dataset is associated with the name of the conflict-of-interest class to which it belongs. The subject is the user in the system. Access to data is constrained by what data the subject has already accessed. All subjects are allowed to access at most one dataset which belongs to a same conflict-of-interest class.

The environment of stock exchange or investment house is a natural environment for this model. Consider the database of an investment house, which consists of company information about investment that investors are interested in requesting. Analysts use these information to guide the companies' investments, as well as those of individuals.

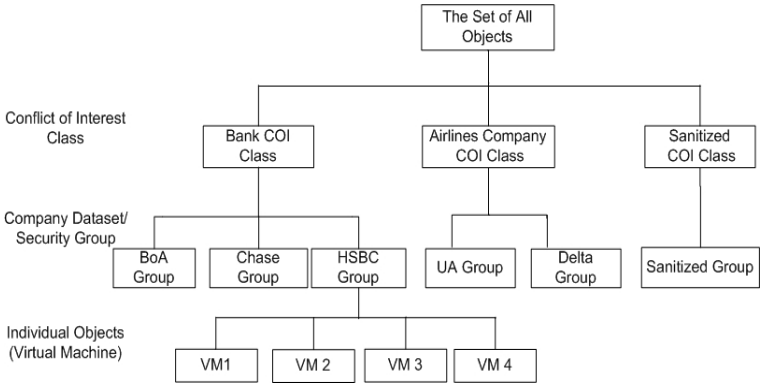



Fig. 2: The Composition of Objects in Clouds

## III. Chinese Wall Security Policy in Cloud Computing

To enforce Chinese Wall security policy in cloud computing for preventing the problems of insecure information flow, several key challenges need to be addressed:

*Challenge 1: Choosing Appropriate Service Layer.* As mentioned in Section I, cloud computing services are delivered at three layers, namely, SaaS, PaaS and IaaS. We need to consider at which layer to enforce the Chinese Wall security policy. SaaS cloud offerings are application-based where more fine-grained access control mechanism is needed. It is relatively complicated and hard to support Chinese Wall security policy at this layer. PaaS cloud offerings deliver services for application developers who usually do not directly deal with end users' data. Hence, enforcing Chinese Wall security policy at this layer is not an ideal approach. IaaS cloud offerings provide physical infrastructure where user's data is stored and processed. Information leaks are more likely to happen at this layer and it is relatively practical to enforce the Chinese Wall security policy at this layer. Consequently, our work focuses on the IaaS layer of a cloud computing environment.

*Challenge 2: Definitions for Policy Components.* For supporting Chinese Wall security policy in clouds, we need to identify the entities in cloud computing environment which correspond with the elements of Chinese Wall security model including subjects, objects, and read/write operations with the formal definitions. Without identifying and formalizing those entities, the Chinese Wall security policy cannot be well incorporated in a cloud computing environment. Since we focus on IaaS layer cloud offerings, we consider cloud virtual machines as objects, cloud service users as subjects and access to cloud virtual machines as the combination of read and write operations.

*Challenge 3: Expressiveness and Effectiveness of Policy Specification.* To control the access to cloud virtual machines for preventing information leaks, we need to define the policy specification. The specified policies will help us determine whether access requests to cloud virtual machines should be granted or denied. Considering that access decision workload based on the policy in cloud computing environment is intensive, we also need to design efficient algorithms to enforce policies.

In the subsequent sections, we first define subjects, objects and access operations in our Chinese Wall security model for a cloud computing environment. Based on these definitions, we then define the specification of Chinese Wall security policy in clouds.

### A. Chinese Wall Security Model in Clouds

Our Chinese Wall security model for cloud computing environments consists of three components: *Subjects*, *Objects* and *Access Operations*. To define the objects, we first need to give the definitions of *Cloud Instance*, *Security Group* and *Conflict-of-Interest Class*.

*Definition 1:* [**Cloud Instance**] A cloud instance is a virtual machine running on the cloud infrastructure. It stores customers' data and hosts various kinds of cloud services. Let *I* denote the set of cloud instances, *I* = $\{i_1, ..., i_n\}$.

*Definition 2:* [**Security Group**] A security group is a named domain containing several cloud instances on an as-need basis [15]. The instances in the same security group are usually dedicated to serving for the same company. For example all the instances in the Bank of America security group store data and host services which are related to Bank of America. Let *G* denote the set of security groups, *G* = $\{g_1, ..., g_n\}$.

*Definition 3:* [**Conflict-of-Interest (COI) Class**] A COI class contains several security groups. Security groups belonging to the same COI class provide services for competing companies. Let *C* denote the set of COI classes, *C* = $\{c_1, ..., c_n\}$.

Based on the above definitions, we give the definition of *objects* as follows:

*Definition 4:* [**Objects**] An object of the Chinese Wall security policy in the IaaS cloud computing environment is a cloud instance. Let *O* denote the set of objects, *O* = $\{obj_1, ..., obj_n\}$ and an object $obj_i \in I$.

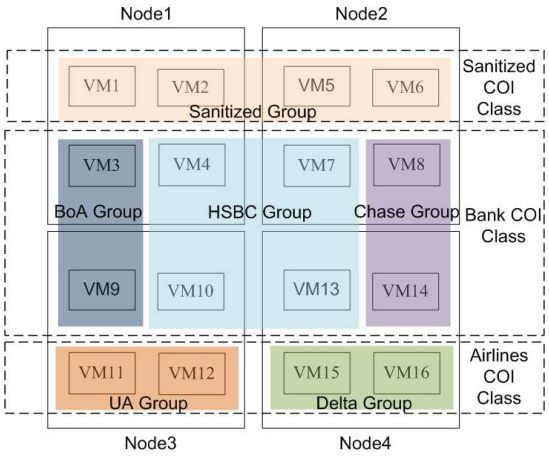



Fig. 3: VM, Security Group and COI Classes

Figure 2 shows the composition of objects in our scenario mentioned in Section I which consists of three levels as follows:

1) The lowest level denotes individual objects and each object is a cloud instance associated with a security group.
2) The intermediate level denotes security groups including BoA Group, Chase Group, HSBC Group, UA Group, Delta Group and Sanitized Group and each security group contains several cloud instances.
3) The highest level denotes conflict-of-interest classes including Bank COI class, Airlines Company COI class and Sanitized COI class.

Note that the sanitized COI class contains a sanitized group which does not have conflict-of-interest issues with any other security group. The cloud instances in the sanitized group usually provide some utility services which do not store or process any customer related data. We denote the sanitized object as $obj_0$. Based on the three levels of objects as shown in Figure 2, we further derive two properties associated with objects:

1) Any two objects which belong to the same security group belong to the same conflict of interest class.
2) Any two objects which belong to different conflict of interest classes belong to different security groups.

We formally define the above two properties as follows:

*Definition 5:* [**Object Properties**]

- *OG* $\subseteq O \times G$ *is a many-to-one cloud instance object-to-security group assignment relation. (obj, g)* $\in$ *OG means an object obj belongs to a security group g;*
- *GC* $\subseteq G \times C$ *is a many-to-one security group-to-COI class assignment relation. (g, c)* $\in$ *GC means the security group g belongs to the COI class c;*
- $O \rightarrow G$ *is a function that maps a cloud instance object to a security group,* $SG(obj_i) = \{g \in G \mid (obj_i, g) \in OG\}$*;* and
- $O \rightarrow C$ *is a function that maps a cloud instance object to a COI class,* $COI(obj_i) = \{c \in C \mid (obj_i, g_i) \in OG \wedge (g_i, c) \in GC\}$*. Therefore, object properties are defined as:*

1) $SG(obj_1) = SG(obj_2) \Rightarrow COI(obj_1) = COI(obj_2)$
2) $COI(obj_1) \neq COI(obj_2) \Rightarrow SG(obj_1) \neq SG(obj_2)$

To better understand how the composition of objects distributes on the cloud infrastructure, we also depict our scenario in Figure 3. There are 16 VMs (cloud instances) running on 4 nodes, each of which is actually a physical machine. Each security group consists of several instances across nodes. Instances 1, 2, 5, 6 belong to Sanitized Group; Instances 3, 9 belong to BoA Group; Instances 4, 7, 10, 13 belong to HSBC Group; Instances 8, 14 belong to Chase Group; Instances 11, 12 belong to UA Group; and Instances 15, 16 belong to Delta Group. In addition, Sanitized Group belongs to Sanitized COI Class; BoA Group, HSBC Group and Chase Group belong to Bank COI Class; and UA Group and Delta Group belong to Airlines COI Class.

Now, we define subjects and access operations as follows:

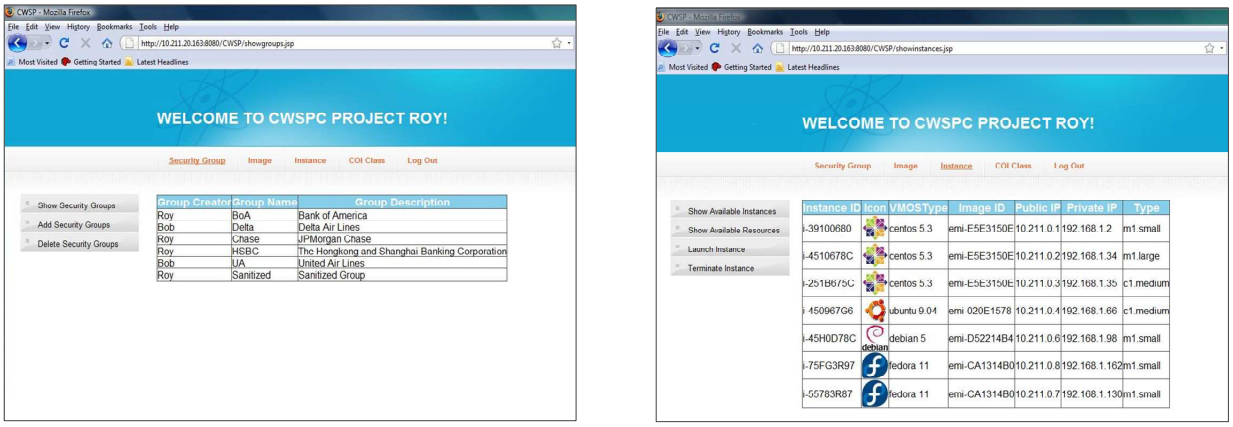
*Definition 6:* [**Subjects**] *A subject of the Chinese Wall security policy in the cloud computing environment is a user who accesses to the data or services hosted in the cloud instance. Let S denote the set of subjects,* $S = \{s_1,...s_n\}$.

*Definition 7:* [**Access Operations**] *An access operation includes reading and writing data and using services hosted in the cloud instance by a subject. Let*

- *ACC* $\subseteq S \times O$ *be a many-to-many subject-to-object access relation. A subject-to-object access relation can be represented by (sub, obj)* $\in$ *ACC, which means the subject sub has accessed the object obj,*
- *ACC* $\rightarrow$ *Boolean be a function that maps a subject-to-object access relation to a boolean value, where*
  - *Access(sub, obj) =* $\{true \mid (sub, obj) \in ACC\}$,
  - *Access(sub, obj) =* $\{false \mid (sub, obj) \notin ACC\}$.

### B. Chinese Wall Security Policy Specification

We also capture policies based on the elements defined above.




(a) Display Security Groups

(b) Display Cloud Instances

Fig. 4: Displaying Security Groups & Cloud Instances

*Definition 8:* [**Policy Specification**] *Let OA is a function mapping each subject to a set of objects, OA($sub_i$) = {obj ∈ O |Access($sub_i$, obj) = true}. Then a subject sub ∈ S can access an object obj ∈ O if and only if any of the following requirements holds:*

1) *There is an object obj' ∈ O such that Access(sub, obj') = true and SG(obj') = SG(obj);*
2) *For all objects obj', obj' ∈ OA(sub) ⇒ COI(obj') ≠ COI(obj);*
3) *obj = $obj_0$.*

*where, initially OA(sub) = ∅, and the initial access request is assumed to be granted.*

The above policy implies that a subject is allowed to access any object in the same security group he has already accessed. A subject is also allowed to access any object in a different security group which is in a different COI class compared with the security groups he has accessed. A subject is freely access any object in the sanitized security group. Considering Alice as a consultant for all the customers mentioned in our scenario, she can access any cloud instance in the BoA group, Chase group, HSBC group, UA group, Delta group and Sanitized group initially. Suppose Alice has accessed a cloud instance which belongs to BoA Group shown in Figure 2. Then she cannot access any instance which belongs to Chase Group or HSBC Group because BoA Group, Chase Group and HSBC Group belong to the same conflict of interest class. However, she can still access any instance in either UA Group or Delta Group, while accessing any instance in Sanitized Group without having any restrictions.

## IV. System Design

Our system architecture mainly contains three layers. The top layer is the system management which consists of six functional components including registration, authentication, security group management, image management, COI class management and instance management. The second layer is the cloud fabric built based on Eucalyptus open-source software, which consists of three components including cloud controller, cluster controller and node controller. This layer interacts with the bottom layer, which provides infrastructure resources.

The original Eucalyptus open-source software implementation of cloud computing does not support a web interface to manage cloud systems but provides a command-line tool called Euca2ools. Our system management module supports a user friendly web interface which largely reduces the cloud administration cost and enforces Chinese Wall security policy to control the access to cloud instances at infrastructure level. Each functional component works as follows:

*Registration*: This component provides registration service for new users. New users need to be activated by the system administrator.

*Authentication*: This component authenticates users by form-based authentication when loging in.

*Security Group Management*: This component provides security group related operations for administrators. Administrators can obtain the group creator information, group names and group descriptions by displaying existing security groups in the system. They can also create new security groups and delete specific security groups. Each security group belongs to a conflict-of-interest class. When administrators create a new security group, they need to choose a COI class for the security group to join in.

*Image Management*: Through this component, administrators can display and delete all the existing VM images in the system. They can obtain image ID information for launching new cloud instances. The supported images contains centos 5.3, ubuntu 9.04, debian 5 and fedora 11.

*Instance Management*: This component provides cloud instance related operations for administrators. Administrators can display current available resource information of the cloud infrastructure which indicates how many instances they can launch in terms of different types of instance scales – small, medium, large, xlarge and xxlarge. Different types of instances consume different resources regarding the number of CPU,

the amount of RAM and the disk size. Administrators can show all existing instances in our system. By doing so, they can obtain instance information including public IP address, private IP address, image ID, instance type, and instance operating system type. They can also launch new instances by specifying image ID, number of instances, and choosing instance type and security group. They can terminate the instance by choosing the instance ID.

*COI Class Management*: This component provides COI class related operations for administrators. Administrators can display, add, and delete COI classes. They can also change existing security groups from the original COI class to another COI class for updating COI classes. This component enforces the Chinese Wall security policy to control the access to cloud instance for preventing information leaks. Users can access instances through SSH connection with a SSH key. A SSH key is generated when a new instance is launched and the instance can be accessed only by this key. Users can download the SSH key from our system. Initially, a user can access any instance with a legitimate SSH key. After a user has accessed an instance, our system will prevent the user from accessing other instances in different security groups but within the same COI class. This component also includes two access decision algorithms for cloud instances that our system dynamically adopts in accordance with the workload of security groups.

## V. Implementation

To demonstrate the feasibility of our approach, we implemented a prototype system called Chinese Wall security policy in Cloud(CWSPC) based on Eucalyptus open-source software v1.6.1. This system is developed using JavaServer Pages (JSP) technologies. We use MySQL Community Server 5.1 for database sever and Xen Hypervisor 4.0 for building IaaS cloud environment. This system facilitates that the SP as a system administrator can manage VM images, cloud instances, security groups and COI classes through our administrative web interface. Figures 4(a) and 4(b) show that an administrator can monitor security groups and cloud instances in the cloud system.

## VI. Related Work

There exist several research work on adopting the Chinese Wall security policy for distributed environments. Minsky [18] proposed the scalable enforcement of a Chinese Wall policy under the inherently decentralized Law-Governed Interaction mechanism. However, the policy enforcement in their approach is decentralized, which is not appropriate for controlling the access to infrastructure resources due to the high implementation and management cost.

Atluri, Chun, and Mazzoleni [6] proposed a Chinese Wall security policy for the decentralized workflow environment, in which they modified the original Chinese Wall policy and enforced read and write rules using restrictive partitions. However, their approach is not scalable enough to support the elastic nature of cloud computing.

Katsuno, Watanabe, Furuichi and Kudo [14] proposed a Chinese Wall Process Confinement offering application-level distributed coalitions with a mandatory access control mechanism for all operating system processes. They implemented a prototype system called ALDC which provides secure operations for office documents on Microsoft Windows even when there are conflicts of interest between the documents. However, their approach is application-dependent and would not be applicable to infrastructure level where we focus in this paper.

## VII. Conclusions and Future Work

In this paper, we first identified the information flow problem which could raise conflict-of-interest issues in cloud computing environments. Also, we have articulated challenges in specifying and enforcing information control policies in cloud computing. To address the identified problem and challenges, we proposed an approach to enforce the Chinese Wall security policy at the IaaS layer of a cloud. We also implemented a prototype system based on Eucalyptus open-source software to prove the feasibility of our approach.

For the future work, rigorous experiments need to be conducted to evaluate the performance of our system. We would improve our approach to support more fine-grained control with generic policy management modules. For instance, we would investigate how IaaS management can be complied with both PaaS and SaaS. In addition, a user may wish to delegate his cloud instance access privileges to others. A practical delegation mechanism is another essential component for cloud computing.

## VIII. Acknowledgments

The work of Gail-J. Ahn, Ruoyu Wu and Hongxin Hu was partially supported by the grants from National Science Foundation (NSF-IIS-0900970 and NSF-CNS-0831360) and Department of Energy (DE-SC0004308 and DE-FG02-03ER25565).

## References

[1] *Amazon Web Services (AWS), Online at http://aws.amazon.com/.*
[2] *Google App Engine, Online at http://code.google.com/appengine/.*
[3] *IBM Business Consulting, Online at http://www/-935.ibm.com/services/us/gbs/bus/html/bcs_index.html/.*
[4] *Saleforce Customer Relationships Management (CRM), Online at http://www.salesforce.com/.*
[5] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, et al. Above the clouds: A berkeley view of cloud computing. *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28*, 2009.
[6] V. Atluri, S. Chun, and P. Mazzoleni. A Chinese wall security model for decentralized workflow systems. In *Proceedings of the 8th ACM conference on Computer and Communications Security*, page 57. ACM, 2001.
[7] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield. Xen and the art of virtualization. In *Proceedings of the nineteenth ACM symposium on Operating systems principles*, page 177. ACM, 2003.
[8] D. Bell, L. La Padula, and M. C. B. MA. Secure computer system: Unified exposition and Multics interpretation. 1976.
[9] D. Brewer and M. Nash. The Chinese wall security policy. 1989.
[10] D. Clark and D. Wilson. A comparison of commercial and military computer security policies. *NIST SPECIAL PUBLICATION SP*, 1989.

[11] T. Erl. *Service-oriented architecture: concepts, technology, and design*. Prentice Hall PTR Upper Saddle River, NJ, USA, 2005.

[12] I. Foster, C. Kesselman, J. Nick, and S. Tuecke. The physiology of the grid: An open grid services architecture for distributed systems integration. In *Open Grid Service Infrastructure WG, Global Grid Forum*, volume 22, pages 1–5. Edinburgh, 2002.

[13] I. Foster, Y. Zhao, I. Raicu, and S. Lu. Cloud computing and grid computing 360-degree compared. *ArXiv e-prints*, 901:131, 2008.

[14] Y. Katsuno, Y. Watanabe, S. Furuichi, and M. Kudo. Chinese-wall process confinement for practical distributed coalitions. In *Proceedings of the 12th ACM symposium on Access control models and technologies*, page 234. ACM, 2007.

[15] T. Mather, S. Kumaraswamy, and S. Latif. *Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance*. Oreilly & Associates Inc, 2009.

[16] J. McLean. The algebra of security. In *1988 IEEE Symposium on Security and Privacy, 1988. Proceedings.*, pages 2–7, 1988.

[17] J. McLean. Security models and information flow. 1990.

[18] N. Minsky. A decentralized treatment of a highly distributed chinese-wall policy. 2004.

[19] D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov. Eucalyptus: A technical report on an elastic utility computing archietcture linking your programs to useful systems. *Computer Science Tech. rep*, 10, 2008.

[20] D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov. The eucalyptus open-source cloud-computing system. In *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid-Volume 00*, pages 124–131. IEEE Computer Society, 2009.

[21] H. Takabi, J. Joshi, and G. Ahn. SecureCloud: Towards a Comprehensive Security Framework for Cloud Computing Environments.

[22] L. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner. A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1):50–55, 2008.

[23] M. Vouk. Cloud computing Issues, research and implementations. In *30th International Conference on Information Technology Interfaces, 2008. ITI 2008*, pages 31–40, 2008.