Implementing XML-based Role and Schema Migration Scheme for Clouds

Gurleen Kaur [1], Sarbjeet Singh[2]
Computer Science and Engineering, UIET
Panjab University, Chandigarh, India
[1] gurleenturka@gmail.com
[2] sarbjeet@pu.ac.in

***Abstract*— Cloud Computing moves computing and storage tasks from individual systems to the Cloud thereby reducing the burden at user's end. It enables the delivery of hardware and software resources over the Internet as a utility. Cloud Computing offers on-demand services with elasticity and scalability on pay per use basis. There are many problems that the organizations face when they migrate their data to the Cloud. Most of the legacy applications are not portable and must be rebuilt in order to fit the target cloud environment. The proposed migration scheme helps the user to migrate the database schema secured with RBAC to the Cloud with much ease. The XML file is generated which carries all the relevant information regarding the roles and database schema of the application. The application at Cloud service provider's end reads the XML file and allocates the space and creates the database schema along with RBAC on the Cloud without much user involvement. The proposed scheme is interoperable and has been demonstrated with a case study.**

Keyword- Cloud Computing, migration scheme, Cloud based access control

I. INTRODUCTION

As Cloud computing is becoming popular and famous, various business organizations are migrating to Cloud. It provides minimum investment for any business to run on Internet. Cloud Computing is essentially composed of a large scale distributed and virtual computing infrastructure. Cloud Computing is a technology which provides computation, software, data access, and storage services that do not require end-user to know the physical location and configuration of the system that delivers the services. This concept can be linked with electricity Grid where end-users consume power without knowing the physical location and infrastructure that provide the service. Cloud offers three service layers to the users viz. Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) [1-2]. Many Cloud service providers are available in the market like Amazon, Microsoft Azure and Google App Engine that offer the user to host applications and store data on the Cloud. Besides providing hosting features, Cloud service providers assist users in the migration process also.

Cloud migration involves moving all or a small part of company's data, applications and services from onsite premises to the Cloud. As Clouds reduce cost and increase service responsiveness, it makes sense for the organizations to shift their business to the Cloud or migrate to the Cloud. Cloud migration initiative may include some risks also for the organization. So a decision is made regarding whether to shift the business to the Cloud or not. This decision is made keeping in consideration various parameters like suitability, maturity and cost benefits etc. [3]. Once the decision is made regarding migration of data to the Cloud, the organization must plan out a strategy before actually moving the business to Cloud. This strategy includes investigating various Cloud migration platforms available to a business, checking out the details of the migration platform and comparing it with the demands of the application etc. This also involves looking at compatibility of the migration platform's internal and external technology with the business application.

The paper is organized as follows: Second section discusses migration and access control challenges. Third section discusses proposed XML-based role and schema migration scheme. Fourth section demonstrates the proposed scheme with a case study and last section presents the conclusion and future scope.

II. MIGRATION AND ACCESS CONTROL CHALLENGES

A. Migration Challenges

One of the major security challenge during migration is the identification and selection of data which needs to be moved on to the Cloud. Before starting the process of migrating the application to the Cloud, it is essential to first identify the business technical factors for the migration. The main aim to migrate the business on the Cloud is to reduce cost and business agility. So it has to be seen whether these factors are maintained after migration. An accurate decision has to be made with respect to moving sensitive data on to the Cloud. The decision is made after analyzing the risks involved, cost- benefit ratio and demands of the organization [4-7]. Following aspects need to be considered while deciding about migration:

Data Migration: It involves identification and selection of data which needs to be moved on to the Cloud.

Application Migrations: It involves identification and selection of applications which need to be moved on to the Cloud.

Role Migration: It involves identification and selection of roles which need to be moved on to the Cloud.

Dependency Migration: It involves identification and selection of dependency software which also need to be moved on to the cloud.

B. Access Control Challenges

Another major security challenge after migration of data on the Cloud is the protection of data from unauthorized access. This can be handled by incorporating access control models at various levels. Traditional access control methods like Discretionary access control, Mandatory access control and Role based access control can be implemented on Clouds also to protect the data from unauthorized access. [8] presents access control model which has been implemented at the API level in Cloud. API is the interface between the user and the Cloud service provider. Here two databases are maintained. The first contains the list of registered users and the second has the role of each user to access the application. The user interacts with the API where he gives his credentials and some attributes like IP address of the machine. These credentials and attributes are matched against the entries of registered users with their domain names in the database. The user could be an employee of the organization which is already registered. After the user successfully logs in, the user's organization and his role are identified in the same way from the second database. Here the access control model taken is the RBAC and two level security has been implemented at the API level [8]. [9] presents access control in Cloud Healthcare system. In this scheme the Task Role Based Access Control is applied on the healthcare system. Here instead of providing access rights to roles directly, the permissions have been provided to tasks. Each role is then provided the task and assigned permissions may dynamically change according to the assigned task. Multiple tasks can be assigned to a role and multiple roles are assigned to a task. According to roles, task selection page is displayed. Both the tasks and roles are maintained by the system administration [9]. [10] presents Ontology Based RBAC Deployment in Cloud. Ontology is a conceptual structure which contains knowledge in a domain and their relationships. It specifies a conceptualization of a domain in terms of concepts and their relationships, which is used to generate a common vocabulary for information exchange without uncertainty. In a domain, multiple possible role hierarchies are defined by ontology systems from different communities. There are two types of ontology templates which are provided: role template and policy template. The role template has the role hierarchy for different domains as described in the role hierarchical model. The ontology is used to build up the role hierarchy for a specific domain. The second is the policy template which contains the access rights for every role in the role template. In this scheme, the users are provided with reference role templates. When a user brings his business on the Cloud, he is provided with reference role templates. Other access control strategies have been defined in [11-13]. Applications of XML tools for enterprise wide RBAC implementation have been discussed in [14]. The suitability of web services security specifications in handling other security requirements have been presented in [15].

The proposed migration scheme is based on the use of standardized XML-based role and database schema file which is automatically created based on the current schema and snapshot of onsite database. This file is then sent at service provider's end where it is interpreted and processed by server side application. The server side application then creates the database and roles as per the information contained in the XML file. Next section explains the proposed scheme.

III. XML Based Role and Schema Migration Scheme

The proposed scheme is for the easy deployment of role based database schema on the Cloud. It basically helps the user, who wants to migrate his database secured with RBAC on the Cloud. In this scheme the authorized users belong to roles and the permissions or access rights regarding the usage of database are granted to roles.

The user is provided with the application interface. Through this application interface the user provides all the relevant information needed for the creation of the database and role-based schema on the Cloud. The information includes the name of various database objects and roles for the application and the permissions granted to various roles. It become very easy for the user to create role based schema on the Cloud as the user only need to specify his/her requirements through an interactive interface without taking into consideration of the technicalities of the migration metadata. The XML file is generated from the information provided by the user which is then sent to service providers end where database and role based access control information is read from XML file and processed. During processing, information regarding database objects and roles is extracted and the new database is created along with roles for which resources are dynamically provided by service provider on the Cloud. Figure 1 diagrammatically represent the migration process.

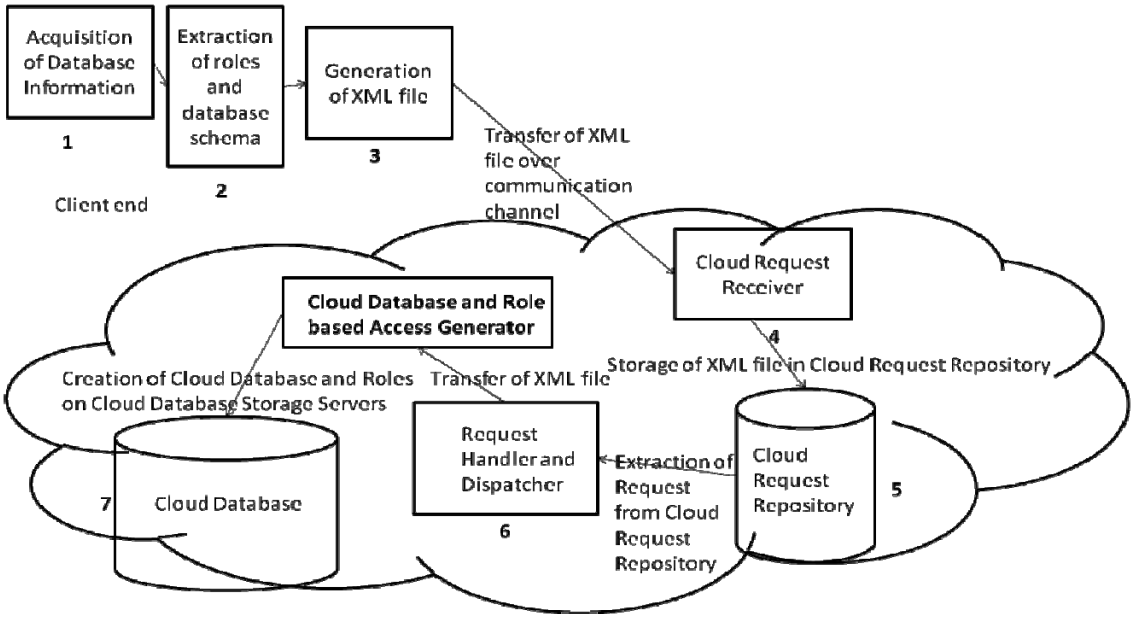

Fig. 1. Design of XML-based Role and Schema Migration Scheme for Cloud

The proposed scheme consists of following steps:

A. Acquisition of Database Information

The first phase of the proposed migration scheme is the acquisition of database information which is required to successfully and automatically reconstruct the database schema of the onsite environment onto the Cloud. The information consists of description of various database objects, relationships, consistency constraints and configuration settings. The user may make use of automated tools that produce the XML document of the entire database schema but in this scheme we have designed an interface which is used by the user to generate XML file which contains all the required information. Though automated tools are useful but they are not flexible. The interface can be customized according to the unique requirements of the organization and provides more flexibility. The designed interface has been used to acquire the database information for the hospital information system application, which has been taken as a case study for the demonstration of the proposed scheme. The details of case study have been presented in section IV.

B. Extraction of roles and database schema

The information regarding roles and database objects is the basis for automatically creating the entire database on the Cloud. The name, type and number of attributes of various database objects are extracted along with roles and access information. Roles describe “who does what” in the application. Access rights are the permissions which are granted to roles to access the database. This information is structured in a standard way by making it comply with the XML schema document which has been specifically designed to hold database, roles and access information in such a way that automatic processing of this information becomes easy for the server application which will handle it at service provider’s end.

C. Creation of XML file

In this step, XML file is actually generated which holds the information gathered during above two steps. This is the file which the user needs to transfer to service provider’s end. As the file contains sensitive information (i.e. information regarding database objects, relationships, access control, roles etc.), it should be transferred in encrypted form over the communication channel to the service provider.

D. Cloud Request Receiver and Cloud Request Repository

The generated XML file is transferred to the Cloud through a secure communication channel. It is received by Cloud Request Receiver which is an application that accepts requests coming from various clients and pools them for batch processing. The file is stored in the database for immediate or later processing. Later processing is required if the service level or other agreements are yet to be finalized otherwise it is processed automatically at the service provider’s end.

E. Request Handler and Dispatcher

Request handler and dispatcher application is responsible for fetching the request (XML file) from the request queue (FIFO) which is in request repository and then forwarding it to another application which processes the XML file according to the information stored in it. The XML file can be given as input to this application manually also but in normal operation it fetches it from the database on first come first serve basis.

F. Cloud Database and Role based Access Generator

This is the component which processes the XML file forwarded by request handler and dispatcher application. During processing, database schema information is extracted from the XML file and various routines are executed which construct the copy of the onsite database on Cloud databases along with roles and their corresponding access rights. The schema information is extracted in the following sequence:

- Extraction of Database Schema: The metadata such as database object definitions, relationships, constraints, roles etc. is extracted from the XML file.
- Extraction of Role Information: The information regarding various roles is extracted.
- Access Rights and permissions Extractions: The information regarding access rights and permissions granted to various roles is extracted.

The allocation of storage space on Cloud is also done at this time. After extracting all the relevant information, the database schema is created on the storage space allocated to the client. Roles are also created along with enforcement of access rights.

Following Section presents a case study of Hospital information System which has been chosen for demonstration of the scheme proposed in this section.

IV. CASE STUDY

The proposed scheme has been demonstrated using an application called hospital information system. The hospital information system taken here keeps the records of patients including the registration details of the patient and the fee payments. As the patient gets admitted in the hospital a detail entry is made of that patient and is updated time to time. The patient who gets admitted is provided with a room depending upon the treatment. The patient is treated by a doctor as per his illness. The doctor may refer the patient to another doctor. When the patient gets discharged then the patient needs to settle the bills sent to him by the accounts department. The health information system also keeps the record of the staff of the hospital. The employee's salary is maintained by the administration manager. All the data of the health information system is maintained in a database. Each employee in the hospital has specific job to do. The job describes the role that employee plays in the hospital. To restrict the access to the database by the authorized users, Role Based Access Control model has been implemented onsite. Different roles considered in this application are:

- Administration Manager: The administration manager has the broadest access to the information including access to personal, financial, clinical and medical information about each resident. The manager is not allowed to prescribe medication to the patient or delete the medical history of the patient. Manager is allowed to add a new resident to the system and keep the data up to date of the patient. Only the manager is allowed to delete the resident who leaves the system.
- Doctor: Every doctor has a unique id, which shows the identification of a doctor and does not match with another doctor. A doctor examines the patient first and according to the problem or disease he/she may prescribe him/her medication. A doctor can refer a patient to another doctor as well.
- Patient: Every patient also gets an id when he/she gets admitted to the hospital. Each patient's data is maintained by the hospital like the doctor who is attending, medication, nurse attending, test prescribed, bills pending etc.
- Nurse: Every patient is attended by a nurse. Nurse takes care of the medication of the patient prescribed by the doctor. Also the test prescribed to the patient being carried out time to time is the duty of the nurse.
- Technician: When the tests are prescribed to the patient by the doctor, those tests are conducted by the technician. After conducting the test the technician submits the report of the patient. Based on the report, doctor gives medication to the patient.

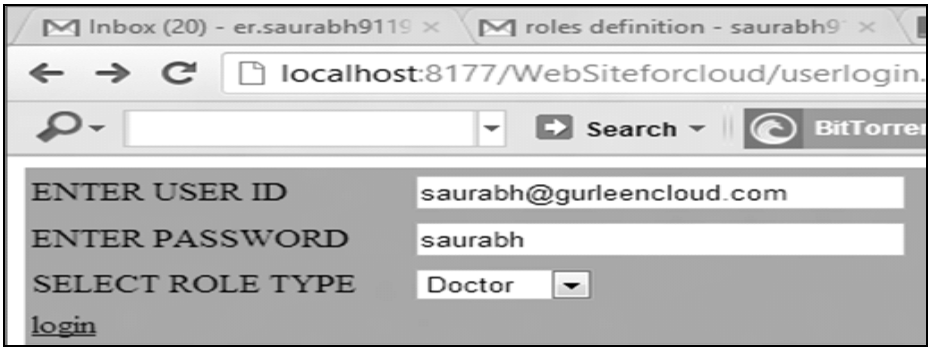

Fig. 2. Screenshot 1 (Login with a role)

The hospital information system described above has been migrated on to the Cloud by making used of the proposed scheme as described in Section III. During migration process, XML schema file is generated which contains information about various database entities, roles and access information. This XML file is transported over secure channel to Cloud service provider's end. At service provider's end, this XML file is processed and database is crated on the Cloud along with roles and access rights are also enforced. Figure 2 and Figure 3 are the snapshots of the system that have been taken when user first logs into the system and when user tries to access some information stored in the database. The doctor then alters the patient id of a patient. The permission to change the id of a patient is denied as it can only be done administration manager.

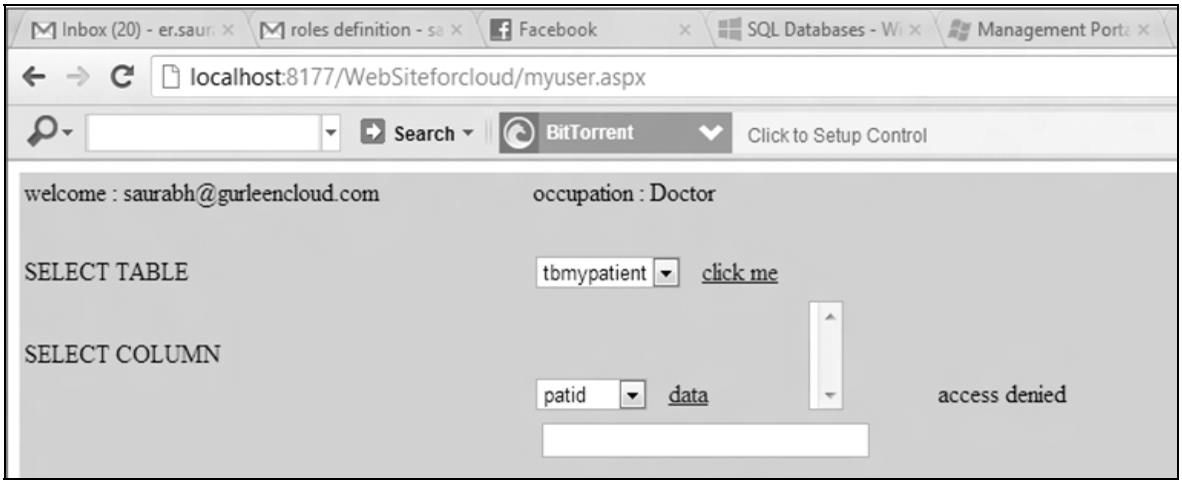

Fig. 3. Screenshot 2 (Access of information by Doctor Role)

As shown in Figure 2, a user can login by presenting his/her credentials along with his/her role. If credentials are valid, access to system is granted and user is then able to perform only those actions for which access is granted to him. Figure 3 presents a scenario where a doctor tries to select some information of a patient for update operation. The access to information for this operation is denied as Doctor Role has not been granted permission to change patient data. Various scenarios of different roles have been considered, executed and evaluated as per actuals. The results obtained show that scheme is functioning as required and hence the approach is workable.

V. CONCLUSION

In this paper, an attempt has been made to propose a scheme that migrate role based database schema to the Cloud. The proposed scheme has been implemented with the help of a case study. The case study has been done on a hospital information system. A hospital information system is migrated to the Cloud by implementing and making use of the proposed scheme. For demonstration purpose we have considered a small part of the application and migrated it to the Cloud using the proposed scheme. In future we are planning to incorporate privacy, trust and access control based models defined in [16-19] in the proposed scheme. While running the application on the Cloud it has been observed that the roles and database schema has been migrated successfully and is working the same way as it was working on-premise.

REFERENCES

[1] G. Tyrone, E. Michael Maximilien, S. Thorpe, and A. Alba, "Towards a Formal Definition of a Computing Cloud", In *Services (SERVICES-1), 2010 6th World Congress on*, pp. 191-192. IEEE, 2010.

[2] T. Dillon, C. Wu, and E. Chang, "Cloud computing: Issues and challenges", In *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, pp. 27-33. IEEE, 2010.

[3] A. Skonnard, and Keith Brown, "An Introduction to Windows Azure platform AppFabric for Developers", *Microsoft Azure Platform.[Online] November* (2009).

[4] S. Krishnan, "Programming Windows Azure: Programming the Microsoft Cloud", O'Reilly Media, 2010.

[5] A. Khajeh-Hosseini, D. Greenwood, and I. Sommerville, "Cloud migration: A case study of migrating an enterprise IT system to Iaas", In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pp. 450-457. IEEE, 2010.

[6] D. Chappell, "Introducing the Azure services platform", *White paper, Oct* 1364, no. 11 (2008).

[7] "Moving from Legacy Systems to Cloud Computing", A Tata Communication White Paper, June, 2011.

[8] A. Sirisha, and G. Geetha Kumari, "API access control in cloud using the Role Based Access Control Model", In *Trendz in Information Sciences & Computing (TISC), 2010*, pp. 135-137. IEEE, 2010.

[9] H. Narayanan, H. A. Jayaprakash, and M. Hadi Gunes, "Ensuring access control in cloud provisioned healthcare systems", In *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*, pp. 247-251. IEEE, 2011.

[10] W. Tsai, and Q. Shao, "Role-based access-control using reference ontology in clouds", In *Autonomous Decentralized Systems (ISADS), 2011 10th International Symposium on*, pp. 121-128. IEEE, 2011.

[11] R. Sandhu, and Pierangela Samarati, "Authentication, access control, and audit", *ACM Computing Surveys (CSUR)* 28, no. 1 (1996): 241-243.

[12] Ahn, G-J, "Specification and classification of role-based authorization policies", In Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. WET ICE 2003, Proceedings, Twelfth IEEE International Workshops on, pp. 202-207. IEEE, 2003.
[13] R. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models." IEEE, *Computer* 29, no. 2 (1996): 38-47.
[14] R. Chandramouli, "Application of XML tools for enterprise-wide RBAC implementation tasks", In *Proceedings of the fifth ACM workshop on Role-based access control*, pp. 11-18. ACM, 2000.
[15] S. Singh, and S. Bawa, "A Privacy, Trust and Policy based Authorization Framework for Services in Distributed Environments", *International Journal of Computer Science* 2, no. 2 (2007): 85-92.
[16] S. Singh, "Trust Based Authorization Framework for Grid Services", *Journal of Emerging Trends in Computing and Information Sciences* 2 (2011): 136-144.
[17] S. Singh, and Seema Bawa, "A Privacy Policy Framework for Grid and Web Services", *Information Technology Journal* 6 (2007): 809-817.
[18] Seema, S. Singh, and D. Sharma, "An Access Control Framework for Grid Environment", *Indian Journal of Computer Science and Engineering*. Vol. 2, No. 6, Dec 2011 – Jan 2012, pp. 937-948.
[19] S. Singh, and S. Bawa, "A Framework for Handling Security Problems in Grid Environment using Web Services Security Specifications", In *Semantics, Knowledge and Grid, 2006. SKG'06. Second International Conference on*, pp. 68-68. IEEE, 2006.