MIGRATION OF APPLICATION SCHEMA TO WINDOWS AZURE CLOUDS

Amandeep Singh[1], Karandeep Singh[2]

[1]Assistant Professor, Computer Science and Engineering Dept, PTU, BMSCE, Muktsar, India
[2]Head & Assistant Professor, MAMC, Agroha, Hisar, Haryana, India

Abstract

Schema migration involves the migration of application schema to clouds. Presently various script generation techniques are there which are used for schema migration, which also sometimes require the application architecture to be changed to make it compatible with cloud platform. This paper shows an approach using user templates for capturing the schema structure and access control mechanism for migration of application schema without using script generation tools. Further it gives the necessary algorithm to show the feasibility of its implementation.

Keywords— *Cloud Computing, rbac, roles, nist, saas, schema, xsd, xml.*

---***---

1. INTRODUCTION

Cloud computing is an emerging computing paradigm in which resources of the computing infrastructure are provided as services over the Internet [1]. Cloud computing is referred to as the next big thing in IT, and a great deal of attention has been focused on cloud technology. In the present scenarios cloud computing is increasingly being adopted by large businesses and mid-sized businesses. Moving to the cloud proves to be a viable option for firms as it dramatically cuts down hardware and software expenditure [2]. For the efficient migration to the cloud, it involves transferring the application data as well as application schema entirely, with authorized access, to the clouds. Migration of the data to clouds is a phased process which requires analysis of the schema which is to be moved, analysis of data and the target cloud platform. BCP is one such application which is used for migration of data but with a major drawback that it cannot be used for the migration of application schema. The challenge is to simplify the process of schema migration and make it more efficient.

2. BACKGROUND

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. It moves computing and data away from desktop and portable PCs into large data centers. Migrating the schema requires capturing the information regarding the table, no of columns in the table and data type of the columns in the table, by providing a user friendly interface. Then, deploying access control over the data requires capturing the information regarding user roles that exist in the system, permissions granted to those roles, according to which data in the tables can be accessed. As the different user roles can be able to access the table data according to the permissions given to the different user roles in the system. In the role based access control

1. A user has access to an object based on the assigned role.
2. Roles are defined based on job functions.
3. Permissions are defined based on job authority and responsibilities within a job function.
4. Operations on an object are invocated based on the permissions.
5. The object is concerned with the user's role and not the user. The Role based access control system is clearly defined and agile, making the addition of new applications, roles, and employees as efficient as possible [3].

3. METHODOLOGY

3.1 Procedure

In this proposed approach 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 information regarding database objects and roles is extracted and deployed over the clouds. For the implementation of the migration procedure following steps are used:

1. Acquisition of Database Information.
2. Extraction of roles and database schema.
3. Creation of XML File.
4. Transfer of XML File.
5. Role based access generation.

3.1.1 Acquisiton 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 proposed approach we have designed an interface which is used by the user to generate XML file which contains all the required information.

3.1.2 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. The use of XML document is to extract access control data from several platforms and represent the access control data for the entire enterprise [4].

3.1.3 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.).

3.1.4 Transfer of XML File and Role Based Access Generation

The generated XML file is transferred to the Cloud by establishing sql connection with database server on the clouds. The 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.

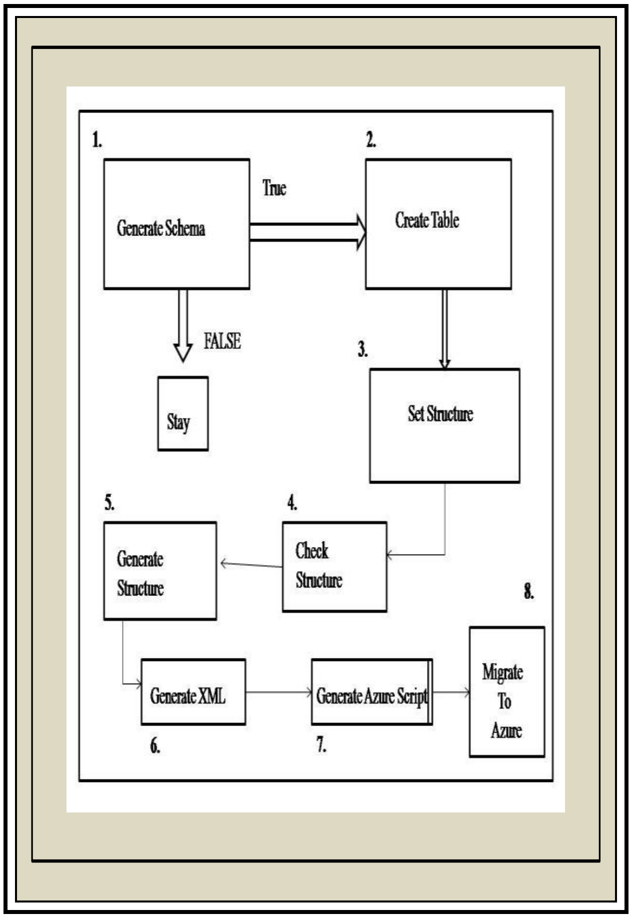

Fig. 1 Steps in the proposed approach

Fig. 1 showing the steps used in the proposed approach.

F. Algorithm Used for generating xml from GUI

The basic algorithm is as follows:

```
1. Get the values of table_count, column_count, role_count,
Permission_count from user.
2. Start with <table> tag // starting of XML.
3. Insert table name from user within  <table_name> tag.
4. Start with the column tag.
5. Insert the column name from user within  <column_name>
tag.
6. Go to step 5 until column_count reaches maximum.
7. End of column tag
8. Goto step 3 until table_count reaches maximum
9. Start with <role> tag
10. Insert role name within <role name> tags from user.
11. Start with <Permission > tag.
```

12. Insert <table_name> <column_name> <read> < write> <delete> tag that the user chooses.
13. Set true/false value within <allowed> tag.
14. End of <permission>.
15. Goto step 11 until permission_count reaches Maximum.
16. Closing tags.

Example scenario:
The proposed scheme is demonstrated using an application called Banking Information System. The Banking Information System taken here keeps the records of customers regarding the registration details of the customers and payments. The employee's salary is maintained by the administration manager. The job describes the role that employee plays in the Bank. To restrict the access to the database by the authorized users, Role Based Access Control model has been implemented onsite.

4. RESULTS

The proposed scheme has been demonstrated by taking example scenario. The case study has been done on a Banking Information system. A Banking Information system is migrated to the cloud by implementing and making use of the proposed scheme.

5. CONCLUSIONS

The approach discussed in this paper can be used for migration of the application schemas to clouds in various real world scenarios. In this research work role based model is used for data access. Further Attribute based model can be combined with the present approach which further enhances the security of access controls mechanisms on the cloud.

ACKNOWLEDGEMENTS

We convey our sincere gratitude to Prof Sandhu M for providing us with the platform and opportunity to work in this area. His encouragement and valuable guidance throughout the course of the paper was a constant support.

REFERENCES

[1]. Preiya, V. S., Pavithra, R. and Joshi., "Secure Role based Data Access Control in Cloud Computing", International Journal of Computer Trends and Technology, pp. 146-151, 2011.
[2]. Rao, R. and Prakash, P., "Improving security for data migration in cloud computing using randomized encryption technique", IOSR Journal of Computer Engineering, vol. 11, no. 6, pp. 39-42, May-June 2013.
[3]. Salunke, D., Upadhay, A., Sarwade, A. and Vaibhav, M. "A survey paper on Role Based Access Control", International Journal of Advanced Research in Computer and Communication Engineering, vol. 2, no. 3, pp. 1340–1342, 2013.
[4]. Chandramouli, R. "Application of XML Tools for Enterprise-Wide RBAC Implementation Tasks", Proceedings of the fifth ACM workshop on Role-based access control, USA, pp. 11-18, 2000.
[5]. Damyanov, I. and Holmes, N. "Metadata Driven Code Generation Using .NET Framework", International Conference on Computer Systems and Technologies – CompSysTech, pp. IIIB.2.1-IIIB.2.6, 2004.
[6]. Das, S., Nishimura, S., Agrawal, D. and Abbadi, A. E. "Database Migration for Elasticity in a Multitenant Database for Cloud Platforms", UCSB Computer Science Technical Report, 2010-09.
[7]. Foster, I., Zhao, Y. and Raicu, I. "Cloud Computing and Grid Computing – 360 Degree Compared", IEEE Grid Computing Environments (GCE08), 2008.