IJCCN | INTERNATIONAL JOURNAL OF COMPUTING & COMMUNICATION NETWORKS

ISSN: 2664-9519 (Online); Vol. 1, Issue 1, August 2019

Perception Reasoning Task-Role RBAC for Data Access Control in Cloud Computing

Abdul Rauf[1], Abdul Hanan Abdullah[1], Saleem Iqbal[1] and Khalid Awan[2]

[1] Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia

[2]Department of Computer Science, COMSATS institute of Information Technology Attock, Pakistan.

* Corresponding Author: Abdul Rauf

Received 12-January; Revised 15-April; Accepted 05-August; Published 15-August

Abstract: Cloud Computing has transformed information and communication technologies into more attractive and advance services. New advance cloud computing systems need safe management environment due to its broad domain. To protect a cloud-computing environment, an efficient access control system is required, which could eradicate malicious outsiders based on its security policies. In this paper, we proposed an access control system, in which the users are assigned their role according to the reasoning mechanism but also classified according to their real jobs. Therefore, every task has a security classification for access and only required permission for completing this task. So this access control is not only ensuring the secure sharing of resources among untrusted users but also support different access permissions for the same user and allowed the user to use secure multiple services. Reasons mechanism is also used to deal with trusted behavior of users according to their access behaviors and give recognition to the user. In the cloud computing system, sharing among various resources take place, which requires a security mechanism for avoiding information leakage and attack of certain intruders.

Keywords: *Task-Role RBAC, Data access control, Reasons to access, time / location based access control and Dynamic SoD.*

1. Introduction

Cloud Computing has become popular and emerged technology to share computational resources including data usage, computing power, data transmission, and mobile communication. This technology is a form of outsourced shared-resource computing in which the users accessed a large range of services. Cloud computing has been used and emerged with various field of life including transportation, healthcare, agriculture, and other monitoring and sharing technologies [1-4]. In cloud computing, various access control models have been used to prevent illegal access of malicious users to cloud computing resources. The main objective of access control models is to ensure the security and privacy of systems [5]. The users can access cloud computing resources and avail services such as computation and storage. The traditional access control models are not able to fulfill the security requirements for the dynamic cloud computing

environment. Therefore, an efficient access control model is needed and particularly significant to ensure security in cloud computing. To meet the security requirements, the user data should have confidentiality, integrity and protection capabilities [6].

The main objective of this paper is to propose an access control system, in which the users are assigned their role according to the reasoning mechanism but also classified according to their real jobs. Therefore, every task has a security classification for access and only required permission for completing the task. The proposed access control model is not for ensuring the secure sharing of resources among untrusted users, but also supports different access permissions for the same user and allowed the user to use secure multiple services.

The rest of the paper is organized as follows: Section 2 presents the related work in the field. Section 3 discusses the proposed access model. Section 4 illustrates the analysis and model advantages. The last section concludes the paper with future direction.

2. Related Work

Various models have been introduced in a cloud computing environment for security provision. In this section, we discuss some existing models and their limitations. There are different prevailing Models are available for access control such as Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role-Based Access Control (RBAC), Task Role-Based Access Control (TRBAC). In DAC, the users can access the resources after undergoing the authorization process, which is based on their identities. In this case, the user identification results in the role assignment that ultimately gets permissions to them. These granted permissions are stored by the system in the access matrix and enable the users to access the resources according to their requirements. However, according to Almutairi, et al. [7], this model may cause of uncontrolled transmission of authority among licensor and the data object which is the cause of insecurity. Moreover, in this model' data flow and the permissions granted to the users which are directed by the owner. Also, it may responsible for information loss and changes in DAC policies due to intruder attack [8, 9].

In MAC, the administrator is responsible to direct certain activities, which are taking place between subject and object and depending upon its policies. According to MAC policies, a user is unable to access resources until the administrator does not direct it. This model is quite fragile because it requires careful planning, and continuous monitoring to keep all resource objects as well as keep user classification up to date. Multi-level security policies are based on required regulation, which is determined by a centralized authority. Therefore, the sensitive data object can only be taken from MAC policy. The MAC mechanism cannot be delegated for permissions, only the main object of fewer descendants and the scope of the license also include the main object of the permissible range. All data processing is based on read up and write down rule [10].

Another model is RBAC, in which the User identification takes place through an authentication process for the assignment of roles to the users. However, for the safe management of the system and to enhance the efficiency of the system it requires a perspective of trust and multi-tenancy in the system at someplace more than one role can be assigned to the user [11]. According to RBAC, roles are assigned to the users after they get permissions by the system to enable them to access resources. These assignments are under the authorization process after the authorized users perform the authorized roles. In a secure environment, information loss can be prevented when the roles assigned to the users are divided either dynamically or statically. However, due to its centralized policies, it may result in permission loss due to non-identification of the privileges. Hence, any malicious user intruding in the system could not be detected by the system [12].

TBAC and RBAC are combined in a new model named TRBAC, which is based on the static and dynamic authorization that connects the role and task respectively. According to this model, the task is firstly assigned by the role before being assigned by the permission for preferable communication between task and object. In TRBAC model, the workflow is based upon the logic unit namely task that could further have sub-tasks and task running is referred to as task instance. These tasks act as the access control centers according to the principle of dependence that could be either sequence, failure or agent dependence. Moreover, convenient processing in the workflow is enabled through authorization step whereas the role is an obligation which is assigned to a certain task which gets permission to the task after satisfying the dependence among tasks [13].

These tasks consist of beginning condition, performance information, and end condition. According to the starting of the activity, which depends upon the beginning condition where the core of the activity is referred to as an information performance. It determines the reason for completing the activity whereas the completion of an activity is ensured by the end condition. The permission is assigned to the user for accessing the system. It could be either the permission of workflow or the non-permission of workflow, which are referred to as the executive and active permissions, which are needed by the task and triggered by the task respectively. Moreover, a session is established by the user and enables it to access multi roles.

TRBAC model is based on five main attributes including user-role assignment, task-role assignment, permission-task assignment, business procedure, and role hierarchy layer. According to these attributes, the relation between the user and its

role, which enables a user to access the multiple roles, which is referred to as a user-role assignment? This assignment is between task and role that enables a task to approach the multiple roles, which is referred to as a task-role assignment. Business procedure and role hierarchy layer present the relation among permission and task that gets permission for the task processing which is referred to as permission-task assignment whereas the running task set and grade relation of roles respectively [14, 15]. Figure 1 shows the TRBAC model.

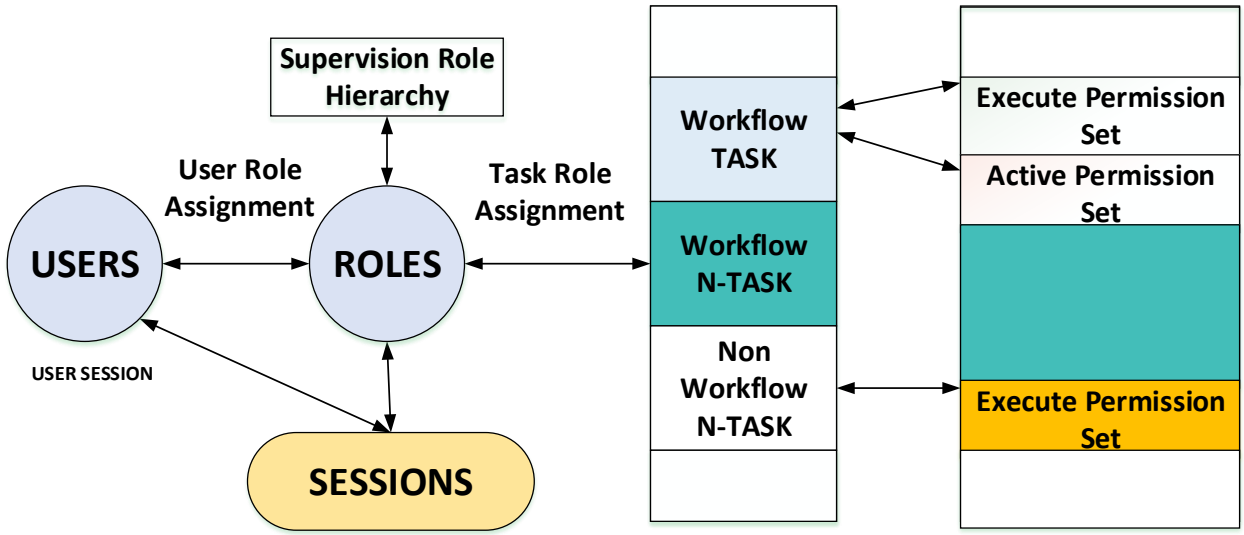

Figure 1. Task-Role RBAC Model

Wang, et al. [16] introduced context technology, time constraints and ACC in the RBAC model for better protection, which is based on the trust relations and authorized certificates. These trust relations may lead to insecurity when the measurement is not following the location constraints. Moreover, Younis, et al. [17] introduced a cloud-optimized RBAC model which is based on granted certifications by the CA. It enables the security of the system through a particular domain under its respective domain manager. Besides this, such a model is unable to follow the heterogeneity principle and unable to maintain security domains and its levels. Later, Sun, et al. [18] provide security for the health care units by introducing semantic access control scheme but due to some security issues this scheme often uses of TRBAC model and enables the active and passive workflows in the environment. On the other hand, Tsai and Shao [19] modified the RBAC model by introducing a reference ontology framework in it, which depends upon the policies for getting permissions from the users for role assignment. Later, ABAC and RBAC were combined by [20] as ARBAC which includes eucalyptus open-source cloud infrastructure consisting of private cloud [21]. After a discussion on related work, the next section introduced the proposed data access control model with its components.

3. Proposed Data Access Control Model

The proposed data access control model involves different roles/tasks, least privilege principle, delegation principle, reason mechanism and security environments for safe management in cloud computing. In the proposed model, identity tags are assigned by the tasks to the users through reason mechanism, which is depending on the dynamic or random behavior of the users and be done indirectly via actual tasks. Besides this, such a model includes certain relations like User Assignment (UA), Role Assignment (RA) and Permission Assignment (PA). According to these relations, the users are assigned by the roles and tasks and get permission for access reasons. Also, these are classified further into the hierarchical order of data tag such as Secured, Concealed, Isolated and Public Promoting S-RH. In the access model, the data tags are the major parts that provide security to the resources in terms of access and depending upon the security of the information. Tasks are further classified after approaching the data tag to access the resources. An identity tag is generated after classification of the task and is assigned to the task through the process to have access over data [22]. The following steps define the basic functions of the data access model.

3.1 Basic Functions:

1. $USER = \{user1, user2 \ldots\ldots, usern\}$ Set of Users
2. $TSK = \{tsk1, tsk2\ldots\ldots, tskn\}$ Set of Tasks
3. $PRMS = \{prms1, prms2 \ldots\ldots, prmsn\}$ Set of Permissions
4. $PRMS = \{prms1, prms2 \ldots\ldots, prmsn\}$
   Such that $PRMS = 2(TSK \times OBJ)$
5. $R = \{r1, r2 \ldots\ldots, rn\}$ Set of Roles
6. $S = \{s1, s2 \ldots\ldots, sn\}$ Sessions Set
7. $Data = \{d1, d2\ldots\ldots, dn\}$ Data Set
8. $SCL = \{\text{Secured (S)} > \text{Concealed(C)} > \text{Isolated(I)} > \text{Public (P)}\}$ Classification
9. $DT = \{dt1, dt2\ldots\ldots, dtn\}$ Set of Data Tag
10. $IT = \{it1, it2\ldots\ldots, itn\}$ Set of Identity Tag
11. $DRS = \{drs1, drs2\ldots\ldots, drsn\}$ Dimensional Reasons.
12. $DRSR = \{drsr1, drsr2\ldots\ldots, drsrn\}$ Dimensional Reasoning with Role
13. $UserAssignment(UA) = User \cap Role$ where
    $UA \subseteq USER \times R$, user to role assignment (many to many mapping)
14. $RoleAssignment(RA) = Role \cap Task$
    $RA \subseteq R \times TSK$, role to task assignment (many to many mapping)
15. $PermissionAssignment(PA) = PRMS \cap Task$
    $PA \subseteq PRMS \times TSK$, permission to task assignment (many to many mapping)

Every user has a numbers of roles $\{r_1, \ldots\ldots r_n\}$, every role has numbers of tasks $\{tsk_1, \ldots\ldots tsk_n\}$, every task needs some permission $\{prms_1, \ldots\ldots prms_n\}$ to complete the job and need classification scl to access the data. The relationship of user to session and task to classification is one to one, and users are not allowed to activate the role outside the particular session. $\forall\ user \in USER \rightarrow s_i \in S$ but $\forall\ tsk \in TSK \rightarrow scl_i \in SCL$, in this case, multiple roles are allowed to user.

3.2 Different States:

1. $tsk \xrightarrow{it} d$. identity tag created for task that wants to access the data, conferring to task classification
2. $tsk_i \rightarrow tsk_j \xrightarrow{itj} d$. identity tag created for task that wants to access the data, conferring to access data indirectly via another task.
3. $tsk_i \xrightarrow{iti} pro \in processes \xrightarrow{itj} d$. identity tag created for task that wants to employ the process, conferring access any process that requires access to data.
4. $tsk_i \rightarrow tsk_j \xrightarrow{iti} pro \in processes \xrightarrow{itj} d$ identity tag created for task that wants to employ the process, conferring access any process that requires access to data and prevent leakage of privileges.

3.3 Context Reasoning Mechanism

3.3.1 Domain

If some space contains one or more object or list of objects, it has called a domain. An example may be an application in fully or partially ordered domain. In multiple domains, create relationship and role represents the participants that provide a grouping mechanism for various jobs. Grouping mechanism is based on job function, which is performed by the user and represents the organizational structure. Therefore, there is a need that space, object, and role all are identifiable by the system [4].

3.3.2 Dimensional Domain

Dimensional domain is a logically restricted and surrounds with either one object or a list of objects and containing dimensional reasons roles, which are called Dimensional Domains (DD). The system must recognize this dimensional domain. Dimensional Domain $DD < DD, DD\_HOP > DD$ represents a dimensional domain name, and DD_HOP is a set of logical locations, which is requiring the area restricted by dimensional domain such as:

$DD\_HOP: occurLDD\ (DD) \rightarrow ll\ \varepsilon\ LL$

A. Multiple Level Dimensional Domain:

Multiple level relationship defines if DD1 and DD2 are two dimensional domains then
Multiple_Level_Domain Logical semantic of the relationship "contain" is define as:

$(DD1, DD2)$ Æ $(ll2, ll2\ \varepsilon\ occur(DD2)$ Æ $(\exists\ ll1, ll1, \varepsilon\ occur(DD1) \wedge contains\ (ll1, ll2)$ contain (Æ ll_1, ll_2)

B. Multiple Dimensional Domains:

Multiple_Dimensional_Domain relationship specified as If $ll_1$and ll_2 be the two location such that
$.ll1\ \varepsilon\ DD1\ \ and\ ll2\ \varepsilon\ DD2$
Multiple_Dimensional_Domain_Overlap

$(DD1, DD2, drs)$ Æ $(\ ll2, ll2)\varepsilon\ occur\ (DD2)$ Æ $(\exists\ ll1, ll1, \varepsilon\ occur(DD1) \wedge overlaps\ (ll1, ll2) \rightarrow overlaps\ (ll1, ll2)$ (1)

Multiple_Dimensional_Domain_Disjoint

$(DD1, DD2, drs)\ )$ Æ $(\ ll2, ll2)\ \varepsilon\ occur\ (DD2)$ Æ $(\ \exists ll1, ll1, \varepsilon\ occur(DD1) \wedge disjoint\ (ll1, ll2)$ (2)

3.3.3 Dimensional Reasons

Dimensional Reasons are the set of reasons in which a relationship is established among the two different domains including set of logical locations [23].

DRSL = {ll1, ll2, ..., lln} where ll ε LL dimensional reasons DRS <drs, drl> drsa is a dimensional reasons name and drsl is dimensional reasons location. The interaction among different domains represent as:

<Insurance, Hospital, Insurance Company {insurance claim, Insurance marketing}>

Similarly, we can define reasoning between multiple domains as:

<Research, Hospital, University {Laboratory Data Analysis}>

Figure 2 shows the response time for dimensional reason role activation.

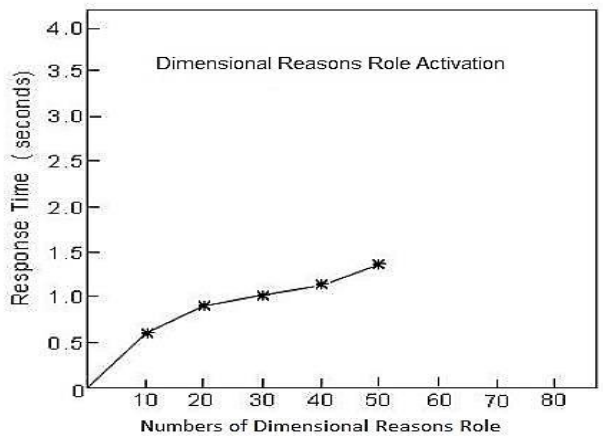

Figure 2. Response Time for Dimensional Reason Role Activation

3.3.4 Limitations:

3.3.4.1 Least Privileged Principle

To complete a particular task, if the subject has extra permission, which is normally, required to fulfill the requirements? In this case, Subject $(subject)$ only need permission $(prms)$ to accomplish its tasks (tsk) to access the data d.

$\forall$ user $\in$ USER $\rightarrow \{r_1, r_2, \cdots r_n\} \forall$ r $\in$ R $\rightarrow \{tsk_1, tsk_2, \cdots tsk_n\}$ and $\forall$ tsk $\in$ TSK $\rightarrow$ $\{prms_1, prms_2, \cdots prms_n\}$

User assign to access role r_i which uses tsk_i *and* only need $(prms)$ to complete the task even when it has extra permission [24].

3.3.4.2 Delegation of Tasks

User collaborate to fulfill their general tasks under the reason mechanism between two subject and work with in same area. $user_i, user_j \in USER, tsk_i \in TSK, r_i \in R$ $user_i$ *assigned* tsk_i, but cannot finished it. An administrator can delegates tsk_i *to* $user_j \leftrightarrow user_i, user_j \in r_i$ and in the same location l_i. So the delegation to tasks to access is supported [14].

3.3.5 Dynamic Separation of Duties (DSoD)

This principal is used to avoid conflict between role and interest, and used to grant more authority to particular user by partitioning tasks, giving permissions related to role [25]. In Dynamic SoD, if user assigned a role x than it cannot be assigned the role y at identical location at any time. That

$SR(user, x, s, ti, l) \Rightarrow \neg(\exists ti' \subset always, \bullet SR(user, y, s, ti', l))$ in case of strong temporal DSoD.

In Dynamic SoD, if user assigned a role x than cannot be assigned the role y at any location and at identical time. That $SR(user, x, s, ti, l) \Rightarrow \neg(\exists l' \subseteq universe \bullet SR(user, y, s, ti', l'))$ in case of strong dimensional domain DSoD.

In single session, if user assigned a role x then it cannot be assigned the role y in the occurrence of similar session. That SR (user,x, s,ti, l)$\Rightarrow \neg(\exists l' \subset$ universe,$\exists$ti $\subseteq$ always $\bullet SR$ (u,y, s,ti$'$, l$'$)). In these cases, for either task and role we used DSoD [26, 27].

$user_i \in USER: r_i, r_j \in R; tsk_i, tsk_j \in TSK$ *where* $r_1 \neq r_j$ *and* $tsk_1 \neq tsk_j$

$user_i$ *activate* r_i *and* $r_j \leftrightarrow r_i \cap r_j = \emptyset$

r_i *assigned to* $user_i, user_i$ *activate* tsk_i *and* $tsk_j \leftrightarrow tsk_i \cap tsk_j = \emptyset$

Data tag is attached to the data identified by the system. Hierarchical ordered set of the data tag is utilized which are used to limit the access according to the degree of security. Classifications of task to access data is Secured (S) > Concealed(C) > Isolated(I) > Public (P). They have to dominate an object data tag before access it [28]. Employ has same hierarchy ordered data tag set to classify the tasks.

$\forall$ tsk $\in$ TSK $\rightarrow$ scl $\in$ SCL, $\forall$ d $\in$ D $\rightarrow$ sl $\in$ SL , tsk access object d $\leftrightarrow$ scl dominate sl. The identity tag is used in this situation especially in untrusted environment

$\forall$ stsk $\in$ STSK $\rightarrow \{TSK_{R \in UA}, slc, USER_{currentlocation}, tsk, RS\}$ RS is (Context Reasoning) [29] and $\forall prms \in PRMS \rightarrow \{Permission\ of\ read, write, execute\ and\ delete\}$. This model supports supervision role hierarchy with strict inheritances. In case where one task depends on another task, categorization of tasks is used to complete a particular job. These tasks are issues an identity tag to prevent any data leakages and conferring to system security, data tag information is passed to application employed by the tasks. Figure 3 shows the data access control model.

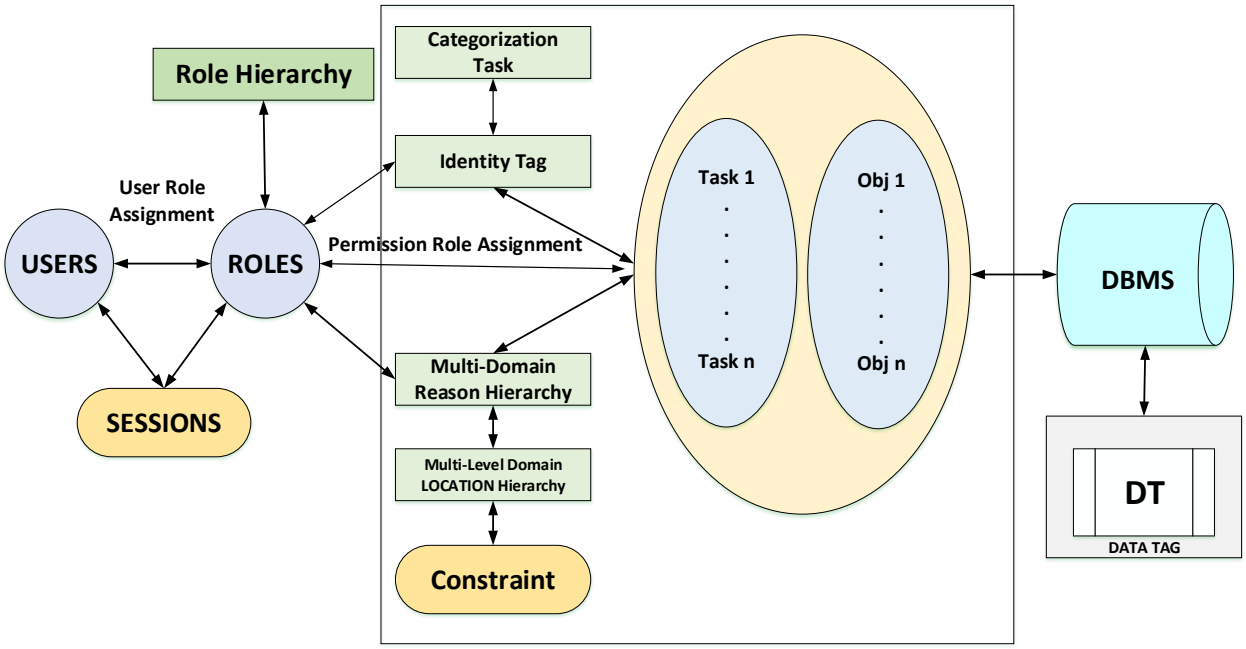

Figure 3. Data Access Control Model

3.3.6 Security Environments

There are two kinds of security environments, comprising of secure and unsecured environments. The secure environment does not require identity tags as only tasks assigned with roles and permissions are granted to users using reasons mechanism. However, unsecured environment utilizes identity tags at each step to maintain security [30].

3.3.6.1 Analysis:

In a cloud computing system, users are assigned by the roles, which are further assigned by the tasks according to the requirement of the users. These tasks carry permissions that enable users to access resources. These permissions have certain classifications and data tag that enable access. To the users, the tasks are assigning under the control of reason mechanism that grants identity tags according to the dynamic and random behavior of the users. The entire above-mentioned proposed security model for cloud is based upon time/location constraints and delegation principles. to fulfill the security requirements. Roles are assigned to the users, which are worth important for the proper working of the cloud computing system and for all the access properties, which are under its processing. In an organization, these roles could be accounting, secretary and manager roles. According to Sandhu, et al. [31] the roles are referred to as job titles belonging to a particular user.

Another major component of the security model is the task that is assigned to the roles and these tasks then ultimately get permissions to the users to enable them to access resources. Each role in the system is assigned with a task that follows an authorization process for the safe management of the system. Permission is assigning to the users through the tasks, which are assigned to them, and depending upon their dynamic/random behavior. These security models are based upon certain constraints and principles that are necessary to fulfill the security requirements of the system. Constraints like time/location constraints and principles like least privilege and SOD are hence important for safe management.

Such models also comprise of certain classifications and tags such as identity tag and data tags, which are responsible for accessing resources and marking of data respectively. This classification is based on the sensitivity of the data lies in a hierarchal order from top secret to unclassified. Hence, these classification and tags are important for a system for proper working as without access to resources is not possible. Another major component of this model is the reasoning mechanism that plays an important role in the safe management of the system. Identity tags are assigned to the users according to their dynamic/random behavior. It is an issue by the reasoning mechanism that is also one of the major parts of the proposed model. These identity tags help the users to access their required resources securely. Figure 4 shows the data access control block diagram.

4. Advantages

In the proposed data access control model, both the tasks and the roles are supporting in active and passive workflows respectively. Besides this, such a model deal with a large number of users according to their trusted behavior while using heterogeneity techniques and imposing certain policies for safe management [32]. The proposed model will secure cloud computing resources with more efficient manners and helpful for service providers and users.

5. Conclusion

The cloud computing environment requires an efficient security model for its safety due to a large number of its users and certain activities taking place in it. However, the existing models are much efficient to support the cloud environment. The existing models comprise the time/location constraints, delegation constraints, and least privilege principle. In the proposed model, the roles being a part of the system are assigned by the tasks that require a classification for the generation of data tag depending upon the security of the data after getting permission to the users for its access. Moreover, users are under the control of the reasons mechanism and are directed based on their dynamic/random behavior. The reasons mechanism being part of the mode, which is responsible to generate identity tags that are required for the safe management of the system. The proposed model is sufficient to provide a more convenient way for cloud computing. In the future, we will work on other aspects of cloud computing.

Acknowledgment

This research is supported by the Ministry of Higher Education Malaysia (MOHE) in collaboration with Research Management Center (RMC) at the Universiti Teknologi Malaysia (UTM) under Vot Number Q. J130000.2528.06H00.

References

[1] K. N. Qureshi, A. H. Abdullah, O. Kaiwartya, F. Ullah, S. Iqbal, and A. Altameem, "Weighted link quality and forward progress coupled with modified RTS/CTS for beaconless packet forwarding protocol (B-PFP) in VANETs," *Telecommunication Systems*, pp. 1-16, 2016.

[2] K. N. Qureshi, A. H. Abdullah, M. Bukhari, and R. W. Anwar, "SSNM-Smart Sensor Network Model for vehicular ad hoc networks," in *2015 International Conference on Smart Sensors and Application (ICSSA)*, 2015, pp. 82-87.

[3] W. Venters and E. A. Whitley, "A critical review of cloud computing: researching desires and realities," *Journal of Information Technology*, vol. 27, pp. 179-197, 2012.

[4] E. Curry, J. O'Donnell, E. Corry, S. Hasan, M. Keane, and S. O'Riain, "Linking building data in the cloud: Integrating cross-domain building data using linked data," *Advanced Engineering Informatics*, vol. 27, pp. 206-219, 2013.

[5] Z. Tari, "Security and Privacy in Cloud Computing," *IEEE Cloud Computing*, vol. 1, pp. 54-57, 2014.

[6] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of network and computer applications*, vol. 34, pp. 1-11, 2011.

[7] A. Almutairi, M. Sarfraz, S. Basalamah, W. Aref, and A. Ghafoor, "A distributed access control architecture for cloud computing," *IEEE software*, vol. 29, pp. 36-44, 2011.

[8] A. R. Khan, "Access control in cloud computing environment," *ARPN Journal of Engineering and Applied Sciences*, vol. 7, pp. 613-615, 2012.

[9] J. B. Joshi, W. G. Aref, A. Ghafoor, and H. Eugene, "Web-based applications," *Communications of the ACM*, vol. 44, p. 39, 2001.

[10] M. Benantar, "Mandatory-access-control model," *Access Control Systems: Security, Identity Management and Trust Models*, pp. 129-146, 2006.

[11] S. Iqbal, A. H. Abdullah, F. Ahsan, and K. N. Qureshi, "Critical link identification and prioritization using Bayesian theorem for dynamic channel assignment in wireless mesh networks," *Wireless Networks*, vol. 24, pp. 2685-2697, 2018.

[12] A. Sirisha and G. G. Kumari, "API access control in cloud using the role based access control model," in *Trendz in Information Sciences & Computing (TISC2010)*, 2010, pp. 135-137.

[13] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," *arXiv preprint arXiv:0901.0131*, 2008.
[14] H. A. J. Narayanan and M. H. Güneş, "Ensuring access control in cloud provisioned healthcare systems," in *2011 IEEE Consumer Communications and Networking Conference (CCNC)*, 2011, pp. 247-251.
[15] C.-x. Zhang, Y.-x. Hu, and G. Zahng, "Task-role based dual system access control model," *International Journal of Computer Science an Network Scurity*, vol. 6, pp. 211-215, 2006.
[16] W. Wang, J. Han, M. Song, and X. Wang, "The design of a trust and role based access control model in cloud computing," in *2011 6th International conference on pervasive computing and applications*, 2011, pp. 330-334.
[17] Y. A. Younis, K. Kifayat, and M. Merabti, "An access control model for cloud computing," *Journal of Information Security and Applications*, vol. 19, pp. 45-60, 2014.
[18] L. Sun, J. Park, and R. Sandhu, "Engineering access control policies for provenance-aware systems," in *Proceedings of the third ACM conference on Data and application security and privacy*, 2013, pp. 285-292.
[19] W.-T. Tsai and Q. Shao, "Role-based access-control using reference ontology in clouds," in *2011 Tenth International Symposium on Autonomous Decentralized Systems*, 2011, pp. 121-128.
[20] E. E. Mon and T. T. Naing, "The privacy-aware access control system using attribute-and role-based access control in private cloud," in *2011 4th IEEE International Conference on Broadband Network and Multimedia Technology*, 2011, pp. 447-451.
[21] Y. Wei, C. Shi, and W. Shao, "An attribute and role based access control model for service-oriented environment," in *2010 Chinese Control and Decision Conference*, 2010, pp. 4451-4455.
[22] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *2010 Proceedings IEEE INFOCOM*, 2010, pp. 1-9.
[23] C. Choi, J. Choi, and P. Kim, "Ontology-based access control model for security policy reasoning in cloud computing," *The Journal of Supercomputing*, vol. 67, pp. 711-722, 2014.
[24] A. L. Pereira, "RBAC for high performance computing systems integration in grid computing and cloud computing," in *Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), 2011 IEEE International Symposium on*, 2011, pp. 914-921.
[25] M. Kandias, N. Virvilis, and D. Gritzalis, "The insider threat in cloud computing," in *International Workshop on Critical Information Infrastructures Security*, 2011, pp. 93-103.
[26] G.-J. Ahn, H. Hu, J. Lee, and Y. Meng, "Representing and reasoning about web access control policies," in *2010 IEEE 34th Annual Computer Software and Applications Conference*, 2010, pp. 137-146.
[27] I. Ray and M. Toahchoodee, "A spatio-temporal role-based access control model," in *IFIP Annual Conference on Data and Applications Security and Privacy*, 2007, pp. 211-226.
[28] S. Pandey, L. Wu, S. M. Guru, and R. Buyya, "A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments," in *2010 24th IEEE international conference on advanced information networking and applications*, 2010, pp. 400-407.
[29] R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed internet of things," *Computer Networks*, vol. 57, pp. 2266-2279, 2013.
[30] B. JAMES, "Security and privacy challenges in cloud computing environments," 2010.
[31] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *Computer*, vol. 29, pp. 38-47, 1996.
[32] R. L. Krutz, R. D. Vines, and G. Brunette, *Cloud security: A comprehensive guide to secure cloud computing*: Wiley Indianapolis, 2010.