A Comparison of Collaborative Access Control Models

Ahmad Kamran Malik
Department of Computer Science
COMSATS Institute of Information Technology (CIIT)
Islamabad, Pakistan

Abdul Mateen
Department of Computer Science
Federal Urdu University of Arts, Science & Technology (FUUAST)
Islamabad, Pakistan

Muhammad Anwar Abbasi
Department of Computer Science
Federal Urdu University of Arts, Science & Technology (FUUAST)
Islamabad, Pakistan

Basit Raza
Department of Computer Science
COMSATS Institute of Information Technology (CIIT)
Islamabad, Pakistan

Malik Ahsan Ali
Department of Computer Science
Federal Urdu University of Arts, Science & Technology (FUUAST)
Islamabad, Pakistan

Wajeeha Naeem
Department of Computer Science
COMSATS Institute of Information Technology (CIIT)
Islamabad, Pakistan

Yousra Asim
Department of Computer Science
COMSATS Institute of Information Technology (CIIT)
Islamabad, Pakistan

Majid Iqbal Khan
Department of Computer Science
COMSATS Institute of Information Technology (CIIT)
Islamabad, Pakistan

***Abstract*—Collaborative environments need access control to data and resources to increase working cooperation efficiently yet effectively. Several approaches are proposed and multiple access control models are recommended in this domain. In this paper, four Role-Based Access Control (RBAC) based collaborative models are selected for analysis and comparison. The standard RBAC model, Team-based Access Control (TMAC) model, Privacy-aware Role-Based Access Control (P-RBAC) model and Dynamic Sharing and Privacy-aware RBAC (DySP-RBAC) model are used for experiments. A prototype is developed for each of these models and pros and cons of these models are discussed. Performance and sharing parameters are used to compare these collaborative models. The standard RBAC model is found better by having a quick response time for queries as compared to other RBAC models. The DySP-RBAC model outperforms other models by providing enhanced sharing capabilities.**

Keywords—RBAC; Collaboration; Privacy; Access control; Security; Information sharing

I. Introduction

User's act of accessing data, information, and resources is controlled to keep check on authorized users and to avoid unauthorized users. Access control is considered as one of the most challenging and complex issues that dynamic collaborative environments face during security administration. The Role-based Access Control (RBAC) model is an approach to control the access of authorized users whenever roles and privileges are involved in a scenario. National Institute of Standards and Technology (NIST) has provided the standard model for RBAC [1]. It has been extended by many researchers to incorporate requirements posed by different applications and scenarios. Collaborative applications are an important research area for access control which tries to control the access of collaborating users. Many different RBAC based models have been proposed for collaborative environments. As such, it appeared that the RBAC model was a good candidate to provide access control. However, a closer examination revealed that although the RBAC model was a good start, additional notions were necessary to effectively apply the RBAC model in a collaborative setting. The first observation was a need for a hybrid access control model that incorporated the advantages of having broad, role-based permissions across object types, yet required fine-grained control on individual users in certain roles and on individual object instances. A second requirement was a need to recognize context associated with collaborative tasks and to apply this context for permission activation. This can be better understood by drawing a distinction between active and passive security models. A passive security model is the one that primarily serves the function of maintaining permission assignments, like RBAC where permissions are assigned to roles. The standard RBAC model is not suitable for collaborative environments because it does not include many data elements that are fundamental for a collaborative environment, such as team, task, user relationships, purpose of access and many more.

The Team-based Access Control (TMAC) model grants more permission based on the team as compared to the Standard RBAC model and works better in a teamwork environment, as the team is the key element in TMAC model [4].

The Privacy-aware RBAC model (P-RBAC) is good in privacy and sharing at the same time because this model implements the privacy policies and uses more data elements to enhance their privacy and sharing due to which this model is better than standard RBAC and TMAC model [3]. This model is more suitable for collaborative environments as compared to the RBAC and TMAC models.

The Dynamic Sharing and Privacy-aware RBAC (DySP-RBAC) model [2] is the best model that works in the most collaborative scenarios, as it introduces more elements (Task, Collaborative Relationships, and Access Level) which are more helpful in maintaining privacy and sharing, so this model is more suitable as compare to other models. This paper selects the DySP-RBAC model which is a collaborative model, evaluate and compare it with the other collaborative RBAC models.

There is always a trade-off between information sharing and privacy. This increases twofold in collaborative scenarios. It is much difficult to quantify who should share how much information with whom in a collaborative system. Access control is normally used to control the access to information. Simply RBAC model does not work in collaborative scenarios where users have collaborative relationships among them which are more granular than roles. For this purpose, RBAC model needs to be extended according to collaboration requirements. This paper focused on aforesaid RBAC models; Standard RBAC model, TMAC model, P-RBAC model, and DySP-RBAC model**.**

The main problem is to identify which model is suitable in the specific scenario by comparing collaborative access control models. The objective of this research is to compare and find the pros and cons of collaborative access control models. This will be very helpful for the researchers who want to use, extend or compare standard RBAC model with their own extensions. Using the comparison of collaborative RBAC models, users will be able to select the best matching model for their application requirements.

This research is carried out to distinguish which RBAC model is better to use for which purpose and in which collaborative environment. It also shows the limitation of standard RBAC model in handling collaboration. Four collaboration based RBAC models are selected, a prototype is implemented for each model and compared the models using metrics selected for performance, access control, and information sharing.

In this paper, Section II describes literature review that explains the background of collaborative RBAC models. Section III is an overview of four RBAC models. Section IV explains the methodology of implementation of four RBAC models. Section V includes results and discussion. Section VI concludes the paper and presents future work.

II. Related Work

In the 1970s, fundamental forms of RBAC were implemented in a variety of ad-hoc forms on many systems. Today’s RBAC derives from the model proposed in [15] and the RBAC model proposed by Sandhu [20]. Ferraiolo and Kuhn also define a formal definition of roles as the set of permissions, hierarchies, subject-role activation, subject-object mediation, as well as constraints on user/role membership and role activation [15]. In 1994, a role graph model for RBAC was developed, by giving efficient algorithms for analyzing role relationships [16]. Ferraiolo, Cugini, Kuhn presented the concept of the separation of duty forms [17]. The family of RBAC models was introduced by Sandhu Coyne, Feinstein, and Youman in 1996 [20] and the method for implementing MAC on RBAC system was also proposed in 1996 [18]. From 1997-1998, Sybase, Secure Computing, Siemens announce RBAC products described as based directly on Ferraiolo-Kuhn RBAC model. The RBAC ANSI standard model was proposed in 2000 [1]. Further, in 2004, American National Standards Institute, International Committee for Information Technology Standards (ANSI/INCITS) adopts RBAC offer as an industry agreement standard.

The concept of the RBAC model is used in different software application and organizations. The purpose of the RBAC models is for management, security and operating system products. This concept is first time introduced in the market as a standard by NIST. This standard is not applicable in every scenario and situation, so it has been extended by many researchers [5, 9, 10, 12, 14]. The RBAC model is very useful in large scale authorization, widely used in many organizations. This model is widely accepted, still, RBAC has some uncertainty and some problems. There are several RBAC based extended models that are used in different scenarios and situations. There are some models like privacy-aware role-based model, team-based access control model and some other models for handling collaborations. Still these RBAC models are not applicable in every scenario and situation.

This research provides a comparison between four RBAC models including the standard RBAC model. Collaborative information sharing environment requires better information sharing among users while privacy laws require for the protection of user’s information from unauthorized access and usage [2]. The DySP-RBAC model is true representative in both domains. A privacy-aware role-based access control (P-RBAC) model is presented in [3]. This model is to force organizations to set privacy policies, privacy framework and enforce the management ideas within organizations. In an organization, there are different kinds of entities like tasks, purposes, relations, and interactions. It can be noticed that in privacy-aware models these kinds of entities are not handled. The P-RBAC model extends the standard RBAC model to express highly complex privacy-related policies, that’s why full-fledged P-RBAC solution is easy to deploy in systems already adopting RBAC, thus allowing seamless integration of access control and privacy policies [3]. There are many more extensions of the RBAC model for handling privacy [11, 13]. The comparison of privacy languages is given in [8]. There is another RBAC extended model TMAC [4] which revolves around teams, where a "team" is an abstraction that encapsulates a collection of users in specific roles and collaborating with the objective of accomplishing a specific task or goal. Users who belong to a team are given access to resources used by a team. Moreover, Collaborative Task Role-Based Access Control (CTRBAC) model [19] and MT-RBAC [7] for the multi-tenants environment to control access to shared resources are available for latest scenarios like Cloud environment. A semantic access control model is also suggested to provide more flexible RBAC for inter and intra-organization environments [6].

III. RBAC Models Selected for Comparison

This section briefly explains four RBAC models that are selected for implementation and further comparison.

A. *The Standard Role-Based Access Control (RBAC) Model*

This is the NIST standard RBAC model to address the core access control issues. This model is organized into four different components Core RBAC, Hierarchical RBAC, Static separation of duty and Dynamic separation of duty.

Core RBAC describes the main aspects of the RBAC Standard model as shown in the Fig. 1. The concept of this model is to assign the roles to the users. Role is a group of permissions; one role can have many permissions. A user can be assigned to many roles and one role can be assigned to many users. The Core RBAC model also included the concept of session in the model. One user can have many sessions but one session is related to one user only. A user can activate one or more roles (that are assigned to her) in a session. A user session tells about the active and inactive roles of that user.

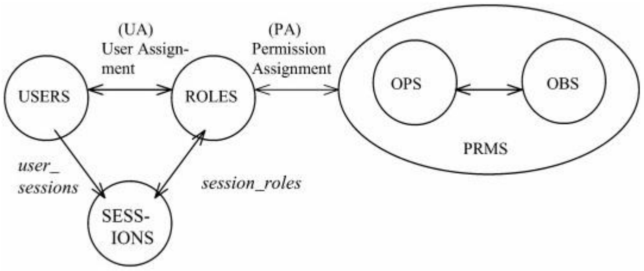

Fig. 1. Core RBAC model [1]

The core RBAC model includes set of elements and their relations. In this model, there are five basic elements that are called the user, roles, objects, operations and their permission. In the RBAC model, users are assigned to the roles and the roles are assigned to permissions. There are many to many relations between user and role, and role and permission. This model also has different kinds of sessions between the user and active roles. A user is assumed to be a human being or any machines or intelligent agents. A role is a job like an employee is assigned to the manager role in the organization and user is fully responsible for the role. Permission assignment is the permission that are assigned to roles for performing an operation on objects. Operations are the set of instructions that execute for the user, for example, in the database system read, write, insert, delete or update.

B. *Team-Based Access Control (TMAC) Model*

This model introduces the concept of team in a collaborative environment by applying the RBAC model. The team consists of a group of users with their assigned role. The team must perform their assigned activity or task. It is a more efficient model because it can assign permission to user in time in a group fashion and support higher degree security. This model plays a very important role in context information related to collaborative activity and can apply this context to decide on permission access. According to TMAC model, the team has two context elements, first one is user context and the second one is object context. The user context is the current user of the team and the object context is the groups of objects that are needed by the team to complete the activities and goals. There are two key directions of the team based access control model. Fig. 2 represents the C-TMAC model components.

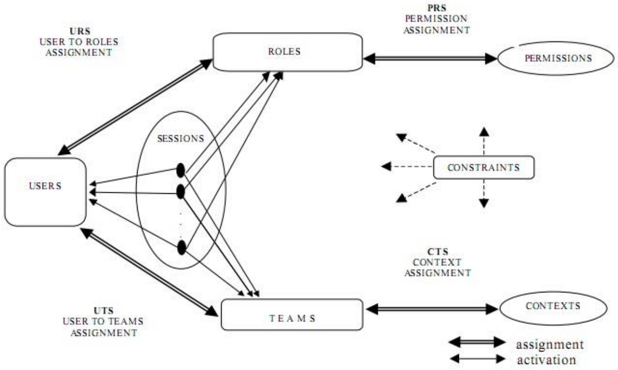

Fig. 2. C-TMAC [4]

Context aware-TMAC (C-TMAC) is an extended version of the TMAC model has five sets of elements that are users, roles, permissions, teams, and contexts. This model also has a set of sessions. It assigns users to the roles and permission to the roles. A team is a group of users, and every user can be a member of one or more teams. There are many to many relations between the role and team through user sessions. This model also has different kinds of sessions between user teams and active roles. Permission assignment is the permission that is given to roles for performing an operation on objects. Permissions are compliance of a particular mode of access to one or more objects. User assignment (UA) and permission assignment (PA) have many-to-many relationships between user-roles and role-permissions respectively. A user can have many roles, and a role can be assigned to many users. Similarly, a role may have many permissions and the same permission can be assigned to many roles. Contextual information examples such as locations and time intervals can be used while granting and denying access. The team theory is used as a system that connects users with contexts.

C. Privacy-aware Role-Based Access Control (P-RBAC) Model

Privacy is one of the important issues in software technology and has received increasing attention from users, companies, and researchers. The privacy protection can only be achieved by forcing privacy policies within an organization. The conventional access models; Mandatory Access Control (MAC), Discretionary Access Control (DAC), and the RBAC model are not made to force the privacy policies and almost not meet privacy and safety requirements. The data collected for one purpose should not be used for another purpose without the approval of the owner of data. The importance of purposes, conditions, and obligations originates from the protection of privacy and personal information. Obligations are the operations to be performed after an operation has been executed on data objects, are essential for some cases.

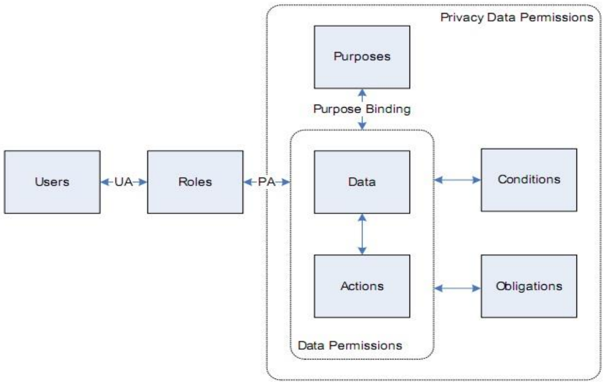

Fig. 3. PRBAC [3]

The Core P-RBAC model has the following set of elements as shown in the Fig. 3: Users, Roles, Data, Actions, Purposes, Obligations, and Conditions. Data are like object. An action is the set of instructions that executes data object for the user. The type of actions depends on the type of system that to be implemented. This model also introduced three new notions purposes, conditions and obligations. In the Core P-RBAC model, permissions are allocated to roles and users get the permissions by being allocated to roles. Conditional access is granted to users using the Conditions data element. Obligations are the conditions that need to be fulfilled after the data access is granted.

D. The Dynamic Sharing and Privacy-Aware RBAC (DySP-RBAC) Model

Collaborative information sharing environment requires better information sharing among users while privacy laws require the protection of user's information from unauthorized access and usage. Keeping this trade-off in view, there is a need for a flexible and better information sharing model that preserves the privacy of user's information.

The DySP-RBAC model extends the RBAC model to integrate sharing and privacy related requirements as shown in the Fig. 4. This model defines the following set of elements: Team, Task, Object, User, Role, Session, Permissions Collaborative relationships, Access level, and three privacy elements; Purpose, Condition, and Obligations. A team is a group of users that performs a specific task. For enhanced sharing, this model defines the sharing elements Collaborative Relationship and Access Level.

In this model action is an executable image of a program that can be used to execute to perform some activity. Permission is an operation allowable on an object. The elements in this model that control the level of data object sharing among collaborating users are Access Level and Collaborative Relationship. Collaborative Relationship element limits the sharing of data objects to only those users who are in a collaborative relationship with each other and Access Level element is used to share only a specific level of information.

The DySP-RBAC model helps in enhanced sharing and is applicable in most collaborative scenarios.

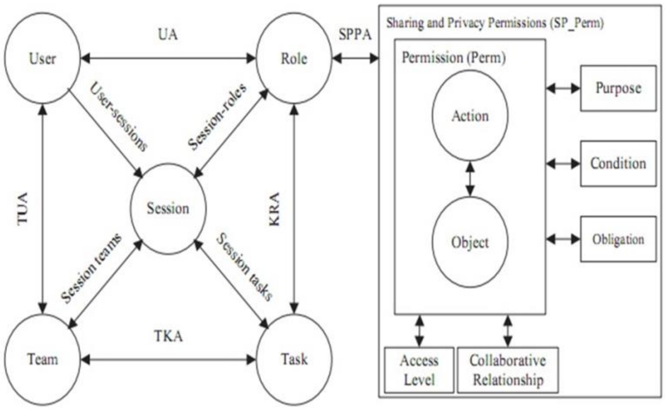

Fig. 4. DySP-RBAC MODEL [2]

IV. Methodology and Implementation

As mentioned earlier, four collaboration based RBAC models are selected for this research and a prototype of each model is implemented using PHP and XAMPP database. Further, these models have been evaluated and compared based on the performance and information sharing metrics. The postman application is used to find out the response time and permissions grants.

Standard metrics for comparison are selected from a list of metrics which are provided by the NIST standard. The metrics are response time, permissions, grants and denial based on several queries. The experiments used the prototype implementations of the RBAC models by comparing the rules and policies for the access control systems. The following parameters are used for the comparison of these collaborative RBAC models.

One of the metrics is performance, which is calculated based on the response time of every model. For this purpose, scenarios are created for each model. Only three data elements, that are common in all models, are selected for comparison of all models using performance metric. These three data elements include role, object, and operation. Using the access control rules based on these three elements, the models are evaluated and compared based on permissions and response time metrics. The permission metric is related to the number of permissions (access rules) relevant to the query and response time metric measures the query response time.

In the scenario for the standard RBAC model, users are assigned to roles. Every role has permissions and user can request for the permission. Permission assignment is the permission that is given to roles for performing an operation on objects. Users, roles, objects, operations, and permissions are defined. For the experiment, 25000 permissions are generated for this model and 25 queries are executed to find out its performance and relevant permissions. For the TMAC model, 25 teams are created in this scenario. Each team has a group of users with their assigned roles. For P-RBAC model, a few more elements are used, those are purposes, conditions, and obligations. Whereas in DySP-RBAC model scenario, there are numerous elements including users, task, team, role, obligation, access level, collaborative relationship condition object, and operations.

Another metric is sharing which is used as a comparison parameter. Using this metric, permission grants of every model are found to check the sharing of collaborative RBAC models. There are 25 queries executed for each model. For sharing metric comparison, the data elements used for each model are listed here. For standard RBAC, only three data elements role, object, and operation are considered in sharing scenario. In TMAC model four parameters role, object, team, and operation are examined. The P-RBAC model uses six data elements including role, object, purpose, condition, obligation and operation. Moreover, role, object, team, task, purpose, condition, obligation and operation are used in the DySP-RBAC model.

V. Results and Discussion

The performance of the four RBAC based models is calculated based on response time, as shown in Table 1. The response time of each model is calculated for an equal number of permissions.

TABLE. I. Queries Vs Response Time

No. of Queries	Response Time of RBAC Models			
	RBAC	TMAC	PRBAC	DySP-RBAC
1-5	1419	1531	1666	1751
6-10	1354	1424	1600	1712
11-15	1418	1512	1605	1764
16-20	1369	1491	1571	1799
21-25	1376	1487	1575	1802

The Fig. 5 shows the response time of all collaborative RBAC models for the sum of five queries. Response time is increasing with the increase in queries. The point to be noted here is that the standard RBAC model has the minimum response time than all other models whereas the DySP-RBAC model has maximum response time than other collaborative RBAC models.

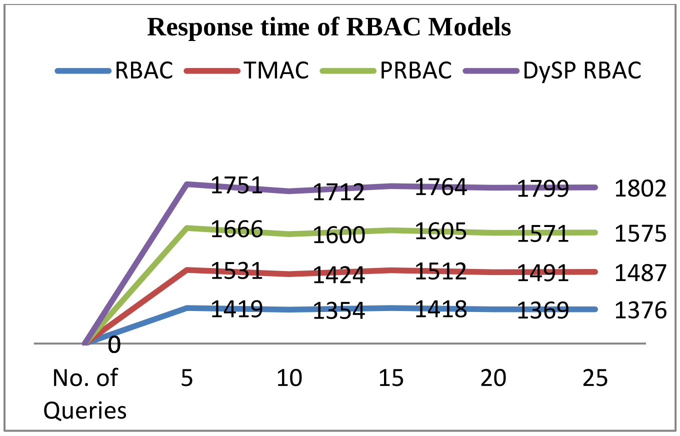

Fig. 5. Response Time Comparison of RBAC Models

Table 2 shows the total running time of 25 queries for all RBAC models.

TABLE. II. Running Time Comparison

RBAC MODELS	RBAC	TMAC	PRBAC	DySP-RBAC
Runtime	6,936	7,445	8,017	8,828

The Fig. 6 shows the response time of all collaborative RBAC models graphically for 25 queries. Standard RBAC model has the minimum running time for queries than other models.

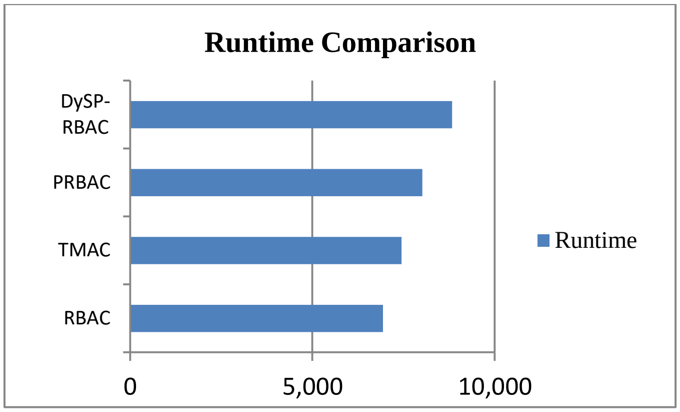

Fig. 6. Running Time Comparison of RBAC Models

The sharing of four RBAC based models is calculated using permissions grant as shown in Table 3. The DySP-RBAC model has the maximum number of permissions grant while executing different sets of queries than all other models, so it can be said that this model is best data sharing model.

TABLE. III. QUERIES VS PERMISSIONS GRANTED

No. of Queries	Permissions Grant of RBAC Models			
	RBAC	**TMAC**	**PRBAC**	**DySP-RBAC**
1-5	835	869	1228	2058
6-10	893	928	1334	2433
11-15	846	894	1284	2338
16-20	821	866	1284	2317
21-25	843	886	1276	2370

The standard RBAC model has the minimum number of permissions grants than other models and DySP-RBAC has granted the most number of permissions for given set of queries as shown in Fig. 7.

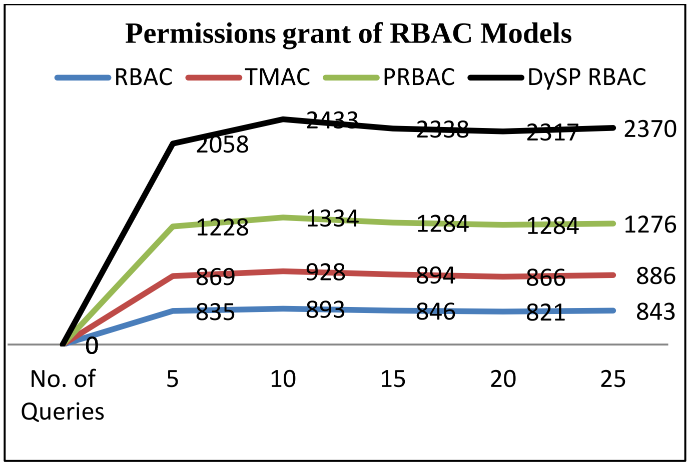

Fig. 7. Permission Grant Comparison

Further, P-RBAC is also good in flexibility for granting permissions than RBAC and TMAC. Even TMAC outperforms standard RBAC in this case. Table 4 and Fig. 8 both represent the permissions granted for all 25 queries for all RBAC models.

TABLE. IV. PERMISSION GRANT COMPARISON

RBAC MODELS	**RBAC**	**TMAC**	**PRBAC**	**DySP RBAC**
Permission Grant	4,238	4,443	6,406	11,516

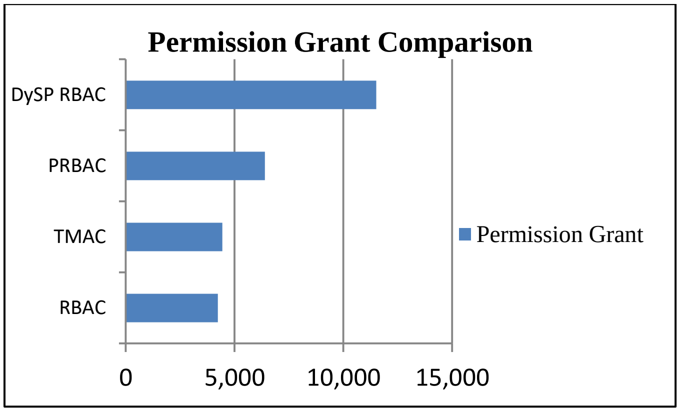

Fig. 8. Permission Grant Comparison of RBAC Models

After the implementation and comparison of four RBAC based collaborative models, it is found out that this research is helpful to explain which RBAC model is better to use for which purposes and in which collaborative environment.

The evaluation of the models predicts that the standard RBAC model is better in performance as compare to other models but less suitable for sharing. The standard RBAC model does not work well in collaborative scenarios. The TMAC model is suitable in the environment where teamwork is involved and can give better performance in sharing as compared to standard RBAC but with more response time. The organizations where the privacy is the key point in sharing data, the most applicable model is P-RBAC model which outperforms standard RBAC and TMAC in privacy and sharing scenarios. The DySP-RBAC model is more suitable in collaborative scenarios and sharing while having maximum response time due to the use of many sharing and privacy data elements. If somebody emphasizes on sharing whatever the response time is, she may opt the DySP-RBAC model. It depends on user requirements and their environment to consider which RBAC model is to be selected.

VI. CONCLUSION

In this paper, four collaborative RBAC based models have been selected for comparison. A prototype for each of these models is implemented and compared based on performance and sharing metrics. After comparison and analysis, it is found that which RBAC model is better to use for which purposes and in which collaborative environment. The performance and sharing of all the models is calculated based on response time and permissions grant. In the end, results are discussed in the form of graphs.

In future, we would like to further work on the RBAC models and try to elaborate their implementation and significance in inter and intra-organizational structures. It would be interesting to provide a complete picture of privacy-aware RBAC models which are more suitable for different collaborative environments. It is also intended to implement an extended version of the DySP-RBAC model for Cloud systems.

ACKNOWLEDGEMENT

This research is funded by *Higher Education Commission (HEC),* Pakistan, through the project *"Information Privacy in Collaborative Environments" approved for CIIT, Islamabad.*

REFERENCES

[1] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, "Proposed NIST standard for role-based access control", ACM Trans. Inf. Syst. Secur, vol. 4, no. 3, pp. 224-274, 2001.

[2] A. K. Malik, and S. Dustdar,"Enhanced Sharing and Privacy in Distributed Information Sharing Environments", in Proceedings of Information Assurance and Security (IAS), Malacca, Malaysia, pp. 286-291, 2011.

[3] Q. Ni, A. Trombetta, E. Bertino, and J. Lobo, "Privacy-aware role based access control", In Proceedings of symposium on Access control models and technologies (SACMAT), New York, NY, USA, pp. 41-50, 2007.

[4] C. K. Georgiadis, I. Mavridis, G. Pangalos, and R. K. Thomas, "Flexible team-based access control using contexts", in Proceedings of symposium on Access control models and technologies (SACMAT), New York, NY, USA, pp. 21-27, 2001.

[5] M.L. Damiani, H. Martin, Y. Saygin, M.R. Spada, and C. Ulmer. "Spatio-Temporal Access Control: Challenges and Application", in Proceedings of symposium on Access control models and technologies (SACMAT), New York, NY, USA, pp. 175-176, 2009.

[6] A. Kamoun, and S. Tazi. "A semantic role-based access control for intra and inter-organization collaboration", in Proceedings of IEEE Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Parma, Italy,pp. 86-91, 2014.

[7] B. Tang, Q. Li, and R. Sandhu, "A Multi-Tenant RBAC Model for Collaborative Cloud Services", in Proceedings of Privacy, Security and Trust (PST), , Tarragona, Catalunya, pp. 229-238, 2013.

[8] A. H. Anderson, "A comparison of two privacy policy languages: Epal and xacml", In Proceedings of Secure web services (SWS), New York, NY, USA, pp. 53–60, 2006.

[9] G. Cabri, L. Ferrari and L. Leonardi, "Agent Role-based Collaboration and Coordination: a Survey About Existing Approaches", in Proceedings of IEEE Systems, Man and Cybernetics, Hague, Netherlands, pp. 5473-5478, 2004.

[10] M. N. Kamel Boulos and S. Wheeler, "The emerging web 2.0 social software: an enabling suite of sociable technologies in health and healthcare education," Health Info Libr J, vol. 24, no. 1, pp. 2–23, 2007.

[11] A. K. Malik and S. Dustdar, "A hybrid sharing control model for context sharing and privacy in collaborative systems", in Proceedings of Information Networking and Applications (WAINA), Biopolis, Singapore, pp. 879–884, 2011.

[12] G. Ahn and R. Sandhu, "Role-based authorization constraints specification", ACMTrans. Infosys. Sec., vol. 3,no. 4, pp. 207-226, 2000.

[13] R.W.Baldwin, "Naming and grouping privileges to simplify security management in large databases", in Proceedings of IEEE Research on Security and Privacy, Los Alamitos, Calif, pp. 116–132, 1990.

[14] D. E. Bell, and L.J.L.Padula, "Secure computer systems: Unified exposition and MULTICS Interpretation", Tech.Rep. ESD-TR-75-306, the MITRE Corporation, Bedford, pp. 1-129, March,1976.

[15] D. Ferraiolo and R. Kuhn, "Role Based Access Control" in Proceedings of National Computer Security (NCSC), Baltimore, Maryland, pp. 554-563, 1992.

[16] M. Nyanchama and S.L. Osborn. "Access rights administration in role-based security systems", in Proceedings of IFIP WG11.3 working conference on database security, Amsterdam, The Netherlands, pp. 37-56, 1994.

[17] D.F. Ferraiolo, J. A. Cugini, D.R. Kuhn, "Role Based Access Control: Features and Motivations", in proceedings of Computer Security Applications, New Orleans, LA, pp. 241–248, 1995.

[18] R. S. Sandhu, "Role Hierarchies and Constraints for Lattice Based Access Controls", in Proceedings of European Symposiums on Research in Computer Security, Rome, Italy, pp. 65-79, 1996.

[19] M.A. Madani, M. Erradi, and Y. Benkaouz, "A Collaborative Task Role Based Access Control Model", Journal of Information Assurance & Security, vol. 11, no. 6, pp. 348-358, 2016.

[20] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. "Role-Based Access Control Models", Computer, vol. 29, no. 2, pp. 38-47, 1996.