# A Hybrid Access Control Model with Multilevel Authentication and Delegation to Protect the Distributed Resources

**V. Nirmalrani[1] and P. Sakthivel[2]**

[1]Department of Information Technology, Sathyabama University, Chennai - 600 119, India.
[2]Department of Electronics and Communication Engineering, Anna University, Chennai-600025, India.

(Received: 29 April 2015; accepted: 01 July 2015)

**The web becomes a common platform to share the resources among a very large group of people. Protecting the resources from malicious users and their actions is a great challenge. Access Control and Authentication are the two major services mostly used for protecting the resources. Many access control models were proposed and standardized by NIST. Some popular access control models are Role Based Access Control, Attribute Based Access Control, Policy Based Access Control, Budget Aware Access Control, etc. became the challenging factors in web for providing access to the resources. Individually those models are vulnerable for web resources. Many authentication methods were proposed earlier for protecting the web resources. These existing methods for ensuring authentication are text based passwords or image based authentication or one time password. Those methods are the most common and widely used methods by many of the real time web applications to verify the authentication of the users. Some common issues in the traditional methods are more time consuming, multilevel authentication for all stored resources irrespective of sensitiveness. Also, these methods are not performing well to meet the challenges. Hence this paper proposes a new method to access the web resources using hybrid access control model with multilevel authentication. Depends upon the type of accessing resources, access polices of more than one access control model have been enabled. For ensuring authentication before allowing accessibility, multilevel authentication for the resources has been fixed based on the level of sensitivity. It facilitates the users to access the resources by consuming short time for authentication process. The proposed system assigns a level of sensitivity for the resources; the sensitivity is proportional to the number of levels which should be crossed by the end user to access the resources.**

**Key words**: Access Control, Authorization, Delegation, Multilevel Authentication, Security, Separation of Duty.

Accessing web resources (information and services) are an essential requirement in many applications. Protecting the web resources from malicious users and actions is a great challenge. For protecting resources from unauthorized and illegal users, authentication method is most important. At the same time, we frequently need to limit or control the access to certain information. Access control policies and methods are used for limiting the access of the resources. Mainly access control is proposed to ensure that the data is accessed only by the authorized people. This makes the system more secure. Access control has three important components: identification, authentication, and authorization. In identification process the user provides the information to tell that the user is authenticated user. The identification information varies according to the different levels of the multilevel authentication system. In the authentication process, the

* To whom all correspondence should be addressed.
E-mail: nirmalrani.it@sathyabamauniversity.ac.in

multilevel authentication method is used to provide more security to the database resources. The resources can be accessed only by the user after crosses the proper level of authentication. Authorization is the process of providing accessibility to the resources. This process actually determines that what can be accessed by the user based on the level of authentication he has passed and also based on access rights provided for the user role. It is done with the preconfigured access rights and permissions provided to the user. The operations of access control are read, write, execute and permit operations. These access right permissions are managed by access control policies. Access control adds an additional layer of protection. To provide more security both access control and multi-level authentication concepts are combined. There are different access control models available. Role and attribute based access control concepts are implemented to provide access security to the resources.

In proposed work a new method which includes multilevel authentication based on the level of sensitivity of resources along with access control is implemented. Sensitivity is proportional to the number of levels which should be crossed by the end user to access the resources. Authentication level is provided according to the sensitivity. Lot of authentication methods like Text based passwords authentication, image based authentication, and one time password, etc., has been proposed to protect the resources.

Role Based Access control (RBAC) is becoming popular in the world of access management; many of the organizations are managing and assigning all access privileges through certain policies for reducing the user misuse capabilities. Many Organization still rely on individual, user-based identity management and individual software applications, however, as the number of users and applications increase, the supporting system becomes time-consuming, infrequent and expensive. Users quickly become frustrated by the need to remember multiple passwords. Hence a low maintenance system that automates routine administration and control access across the networks is needed by the users, so that the data security is ensured while RBAC can be challenging in design and implementation. Low maintenance costs and increased efficiency are the key benefits of RBAC.

Role Based Access Control has different roles of users, access decision of the resources are taken based on user role. The three main components of RBAC are users, roles and permissions. RBAC associates permissions based on role rather than the individual user. It provides access to resources based on the collection of rights and permissions assigned to roles. Users can also have one or more role. Access rights and permissions to access the resources vary based on roles. RBAC simplifies the authorization process because same permissions are given to all users in the same role. Dynamic separation of duty is not included in RBAC. Role permissions are given through a role hierarchy and the permissions are needed to perform the allocated task within an organization. There may be different of inclination between the actions will appear. The divergence may appear when the user wishes to blow up their own self-absorption. This becomes the reason for the users to get personal benefit from misusing the permissions.

So Attribute Based Access Control (ABAC) is used. It tells which role can have access to which attributes. In ABAC access decisions are based on the attributes of users in the role. In ABAC one role can have 2 different access permissions based on the attribute value provided by that role. Some resources can be accessed only by satisfying roles and some resources may be accessed by satisfying both roles and attributes. By combining both RBAC and ABAC dynamic separation is achieved and the system is more flexible. Centralized control over the access of resources is overcome by combining RBAC and ABAC.

BARBAC (Budget Aware Role Based Access Control) is motivated by the imperfection of RBAC when a user's misuse the resources allocated by the administrator. This paper proposes a model to overcome this misuse capability of accepting an existing RBAC policy as an allusion to segregate the price of permissions for users. Through this, those users who based on an existing RBAC policy and permission would pay a base price for permission, while others pay an escalated price. In pursuance of payment for accessing, the users are given a defined budget, allocated according to the administrator's current knowledge of each user's operational needs. A user's capacity to execute tasks is limited by their defined budget. For each access, the corresponding budget will be reduced by the administrator of the database. BARBAC has been implemented with two constraints which are provided as budget for each user. The user can access the permission that has not been already assigned to the user through escalation, the escalation limit, zone, times are defined by the administrator and the administrator manages the overall workflow.

## Related Works

Salim, Jason, Dulleck, Dawson et al (2013) "Budget Aware Role Based Access Control" focuses on Role Based Access Control with Budget notation. In this paper the user should pay for each and every access based on the role allocated by the administrator, the total weight of the role to be calculated as the sum of the cost associated with each and every task.

Simple text based authentication is not secure. Biometric authentication is expensive. To overcome drawbacks from these above methods, Two Level Image Based Authentication System (T-IBA-S) is implemented by C. Thenmozhi, S. Sathvi, B. Thamotharan (2013) because use of the image is more effective and secure. In first level users are asked to select images from the list and are asked to select co- ordinate position in that image. In second level multiple images are asked to select from the provided image set. Time consumption is more and the system doesn't consider the sensitivity of resources.

To provide more security to resources Surabhi Anand, Priya Jain, Nitin and Ravi Rastogi (2012) implemented a 3 Level Security System in which the current level should be passed in order to proceed to the next level. Level 1 is a text based password. Level 2 is image based authentication in which user should select images from image grids. Level 3 is OTP which is sent to mobile or mail. Each resource can be accessed after crossing all the three levels. In this paper sensitivity level of resources and time taken to access resources are not considered. Because of this, more time is consumed to access very less secure resource.

Persuasive Cued Click-Points method is implemented by Sonia Chiasson, Elizabeth Stobert, Robert Biddle, Alain Forget; Paul C. Van Oorschot (2012) to overcome the attacks in text based password authentication method. Persuasive cued click-point system provides more secure by using graphical images as passwords. While registering, users are asked to select, click-points in the provided view port area of the image. The view port is dark and the remaining areas are shaded during registering. While logging the registered image is displayed normally and users are asked to select the same click points as during registering process. This system overcomes more attacks compared to previous methods.

Salim, Reid, Dulleck, Dawson et al (2011) "An approach to access control under uncertainty" deals with balance the security and information availability handled by an approach to access control under uncertainty, In this paper value of resources are explicitly defined and RBAC policy is only used as a reference point to determine the price to pay for access and allocate budget to the user, the user can gain unassigned permission while escalating their permission.

Role based access control (RBAC) model is used to express access control policy and reduce the security administration. This model uses user-role assignment concept. It means assigning policies to roles instead of each user. In large organizations, there will be more roles. RBAC model is a Centralized administration. Scalability is not achieved by using the RBAC model. So the administration task should be distributed to achieve scalability. To decentralize the administration task Administrative RBAC model is used by Amit Sasturkar, Ping Yang B, Scott D. Stoller, C.R Ramakrishnan (2011). This model provides ease of administration and scalability, but understanding this model is difficult.

A lot of information is present in Internet of Things (IoT). To compute and process this information securely Guoping Zhang, Jing Liu (2011) implemented workflow-oriented attribute based access control. Here access control decisions are made based on certain attributes like user attribute and resource attribute. Based on these attributes authorizations were provided. This model manages permissions dynamically when compared to RBAC.

Liu D, Camp LJ, Wang X, Wang L et al (2010) "Using budget-based access control to manage operational risks caused by insiders" alleviate the insider threat. A problem appears when the user agrees to misuse the privilege, which is assigned a budget to each access reduce the risk caused by the user. Assign a price for access based on the behavior of the user Each Access right of a user may cost him in certain risk points. If the user finished his risk budget before completing the task, a penalty will be given in the form of punishments otherwise penalty will be given in the form of reward. It mainly focuses on reducing the risk caused by accessed exception.

Zhao X, Johnson ME, et al (2010) "Access governance: flexibility with escalation and audit" ensure flexibility and security of RBAC and it provides more data base access and reduce the control cost. It also ensures the dynamic nature of the system. Distinct model has also been proposed to improve the compliance of the RBAC model.

Ma X, Li R, Lu Z et al (2010) "Role mining based on weights" explain the role will be mined based on the weight of permission assigned to the user, similarity between the user and permission will be calculated for assigning the role it is a very easy method for finding frequent permission set. The excellent allotment of access permission finds to be very complex while proceeding.

Ebru Celikel, Murat Kantarcioglu, Bhavani Thuraisingham and Elisa Bertino et al (2009) "The Risk Management approach to RBAC" employed about Risk Analysis and Risk Control while accessing the database based on the Role FMEA model used to analyze the effect of assigning the risk priority number and this paper mainly concentrate on user's risk and providing security to the distributed database.

In a large organization, there will be millions of users. Simple manual user-role assignment concept is difficult in large organizations. To overcome this drawback Mohsen Saffarian, Qiang Tang, Willem Jonker, and Pieter Hartel (2009) implemented dynamic user-role assignment concept.

Qun Ni, Alberto Trombetta, Elisa Bertino, Jorge Lobo et al (2007) "Privacy aware Role Based Access Control" proposed the model to detect the conflict between two permission assignment. In this paper Privacy policy permissions and rules are assigned to role in detecting the conflict between the permission assignments.

In context aware environment access control system is used. The simple RBAC model is used first. As this model is static and restrictive in some situations Lorenzo Cirio, Isabel F. Cruz, and Roberto Tamassia (2007) borrow some ideas from ABAC model. In this model permissions are not associated directly with attributes instead it borrows concepts from RBAC. In this paper both access control models are combined and dynamic assignment of roles to users are achieved.

**Problem Definition**

In all the existing works, different authentication method is used to secure resources. But it provides the same level of authentication to all the resources such as less secure, moderate secure and high secure resources. Existing systems is not resource centric. It doesn't consider the importance or sensitive level of resources. So time consumption is more to access very less sensitive resources. Likewise, all the existing work provides authorization either using one of the access controls RBAC or ABAC. RBAC separately has more drawbacks. It has centralized control over resources and scalability, which is little difficult.

In this paper Multi Level Authentication System (MLAS) is implemented. Multilevel means 3 different levels of authentication is used to secure resources. To overcome the drawback of time consumption while accessing resources in the existing system, MLAS is implemented by considering the sensitivity of resources. Sensitivity is proportional to the number of levels which should be crossed by the end user to access the resources. Each resource has different levels of authentication.

Level 1 - Text Based Password Authentication – Registered username and password should be provided by the user with this level of security. Level 2 - One Time Password Generation (OTP) – In this level of security the OTP is sent to the user mail id and the user is asked to provide that OTP for accessing resources. Level 3 – Image Based Authentication – In this level of security, the user is asked to select the click points in the provided image similar to the user selected during registration.

To overcome the drawback of RBAC for authorization process, proposed system uses both RBAC and ABAC. Some resources require only RBAC but some other resources require both access controls for authorizing the users. By this way the system is more scalable and dynamic user role allocation is achieved. Dynamic separation of duty is obtained by ABAC.

**More Time Consuming**

The existing system consumes more time to mine the role, the role will be mined based on the weight of the permission, the weight will be calculated by finding the similarity between both the users and permissions from the similarity matrix. Some mining algorithm can be used to generate the role. It uses top down or bottom up strategies so it takes more time for role assignment. In the proposed system the role will be allocated based on the privileges or priorities of user in the enterprise so it is less time consuming method when compared to existing systems.

**Weak Authentication**

The existing system is using the weakest authentication mechanism. It simply prompts for a user name and password credentials for the end user. These credentials are transferred over HTTP to the server. Some of the e-governance applications encode the credentials using certain hashing methods. Some transfers the data over SSL. This mechanism fails when the password is leaked. Someone who came to know the password either by sniffing or by some other mechanisms, it is very easy for him to tamper the system. Also, in this mechanism, no one can claim that an action was actually done by the same user. In the proposed work, secured word will be used after the user name and password login. Each and every time it will ask the random position of the secure word, so it can be used to avoid the unauthenticated access.

**Redundant Permission Assignment**

In privacy aware role based access control, privacy policies are assigned to each role due to this privacy rule, there may be redundant in role permission assignment. It is very difficult to handle the user access permission. In the budget based access control, budget is assigned for each access so there is no redundancy in user permission assignment.

**Permission Misuse**

Role based access control provides enough access permissions to each and every user, even though the policies are correctly specified the authenticated people can misuse the assigned permission, for example the administrators have the authority to access the database details, some time they may be changing the details for some reasons, misuses will be made by authenticated people. In budget aware role based access control the user should pay for each and every access so the misuse was bounded by allocating budget to every role for each access.

**Scope of the Proposed Work**

The main objective of the proposed solution is to be automating the various functions and activities of the bank through the internet. The solution will facilitate to the bank employees and the account holders with the different modules. This solution is very much necessary for the private sector banks and the corporate sector. The banking industry will take a new shape and explore like never before. Using the solution the bankers and account holders can generate various kinds of reports and reduce the misuse made by the authenticated user. The BARBAC system is defined in terms of Authentication, Authorization and Access Policies (Role and Cost). Many of the existing RBAC are constructed using certain policies and attributes; the information from the database is accessed through multiple level of security. Authorized users only can do the task allocated by the administrator. So the user needs enough access to perform their task, the administrator has to allocate each user the level of access required for their job. By providing suitable security policies the permission will be assigned to each user.

**The Key Elements of RBAC are**

- Users—By definition, users are individuals who perform a job function within an organization. Users traditionally have been designed to perform individual functions within an organization.
- Roles—In a business context, roles represent job functions and related responsibilities. Responsibilities represent users' implicit or explicit authority to execute their job function. In a technological context, roles represent a collection of entitlements that a person inherits from an application perspective to perform a job function.
- Permitting—In a technological context, permission is the provision of authority to someone to perform an operation against an RBAC-controlled object within an application or system.

A certain entity is bound to the access provided by the role they are in. More often than not there are exceptions in the access needs of an entity. It would be rare that very large groups of entities would all need the exact same access. So

- Excellent allotment of permission assignment finds to be very complex.
- More access may lead to increasing risk of misuse the assigned permission.
- The main contributions of this paper are,
- To allow only authenticated users to access the resources, the multilevel authentication system is implemented based on the sensitivity of information.
- To authorize the users based on access control models for protecting the resources.
- To construct Role Based Access Control (RBAC) Framework.
- To construct Attribute Based Access Control (ABAC) Framework.
- To construct Hybrid Access Control (HAC) by combining RBAC, ABAC and BARBAC.

The Detail overview of the proposed Clustering Algorithm is given in this section. A Detailed description of the architecture and the algorithm formulated are stated.

**Proposed System Architecture**

Here the new user should register to access resources. The registered users can access resources by crossing multiple authorization and authentication stages. This authorization level and authentication level differs for different roles of user in the system. There are two levels of authorization. One is role based authorization, and another one is attribute based authorization. Some roles should cross both or either one authorization level to access the requested resource. There are three levels of authentication in this system. This authentication level differs based on the role of user and also based on the sensitivity level of resources too.

This paper also proposes a different budget based approach for RBAC, access choice is based on whether the user can afford the cost of permission, each and every role have different budget, cost is assigned depending upon the role of the user. Each role has different weight, Role weight corresponds to the cost of its associated task. User can able to escalate their permission for access. In proposed framework budget to be referred as the time and task limit associated with each user. A certain cost should be paid for each and every access in the budget based RBAC, so the limit will be provided for each and every access to reduce the misuse made by the authenticated user. Flexibility of RBAC is maintained using permission escalation. Unwillingness to misuse the permission associated with the role.

**Overview of System Architecture**

Role Based Access Control: In RBAC each user has set of roles that are assigned during a session. The user can activate or deactivate any of those roles through the session. The permission available to the user is the permission assigned to the role. Each session associated with a single user and each user is associated with one or more session. The administrator has the responsibility to assign the role; the user should satisfy the assignment policy. If satisfied the role and the permission are being allocated to the user.

**Budget Allocation**

Administrator allocates budget to each user, user should finish their job within the time. Role is having a logical group of task user should satisfy particular role to perform task. Some task requires a complex group of operation. Each user should have a minimum budget to get the access permission, user with allocated role and cost of operation can access the resource. Cost differs for different roles to access the service.

**Database Access**

The database is a highly secured, administrator can only create, delete and update the table. The login details such as password and security word stored in the database will be in the encrypted format. Security will be provided in the form authentication and authorization. The user should pay the cost for executing task through role when the user registers their details; administrator sends the budget details in the mail. The user with allocated budget and time limit can access the database and utilize the permission.

**Delegation**

Delegation is the act of executing tasks through a role that has not been already assigned to the user. Delegator is a person who is authorized to act as representative for another. The delegation will be done by the administrator. The delegation region, limit is assigned by the administrator. The encrypted block of information or actions of the service provider is called Delegate Token, it can be provided by job invocation time or job submission time. Figure 1 and Figure 2 shows the detailed architecture of the proposed work.

The misuse capability of the users can be restricted because of Budget based Access Control. In case of any emergency the concept of delegation can be adopted by the users to do the transactions without having the budget. More people have been served in less time. The resources have been utilized properly and effectively by optimizing the number of transactions through the budget. The administrator can easily search a record and update it if is required in an efficient way. The problem that consigns in this paper is how the budget for each user is allocated by the administrator. Initially the administrator can assign roles for each user with a limited budget, for each and every access corresponding cost will be reduced from the user budget. The time stamp also provided with a budget for each access. The concept of sessions in RBAC which enables users to activate only those roles necessary to complete their jobs. If the user have active role must be authorized to access, Budget can be considered to ensure the completion of the user's task within the time. Each user should have a minimum budget to get access permission, user with allocated role and cost of operation can access the resource, cost differs for different permission and role.

### Proposed System Implementation

### Registration Phase

The proposed solution for multilevel authentication has been implemented in university database resources. There are many different roles in an organization. The registration phase differs for each role. To provide correct registration phase, the role of the requesting user should be found by the administrator. This task can be done using ABAC. There are some unique attributes for each role. By providing values for that unique attributes the role of the user is identified by the administrator. According to the role identified the corresponding registration phase is provided to the user. This task is done by RBAC.

### Login Phase

In login phase level 1 authentication is used. Users are asked to enter the registered username and password. Using the information provided by the user the role allocation has been performed by RBAC. Each role has different access rights. Based on the role identified corresponding access rights are provided to access the resources.

### Access Control

Access control models are created to enforce the rules and objectives of an established security policy and to dictate how subjects can access objects. There are two models that will be covered in this section: Role Based Access Control (RBAC) and Attribute Based Access Control (ABAC). Access control is more than simply requiring usernames and passwords when users want to access resources. There are multiple methods, techniques, technologies, and models that can be implemented, there are different ways to administer controls, and there are a variety of attacks that are launched against many of these access control mechanisms.

There are three important components of access control: identification, authentication, and authorization. Identification is the activity of the subject supplying information to identify itself to an authentication service. Some examples of identification mechanisms are username, account number, and memory card. Authentication is the second part of a credential set to verify the identity of the subject. These mechanisms could be passphrases, passwords, cryptographic keys, PIN numbers, or tokens. Authorization is the process of determining what this identified subject can actually access and what operations it can carry out. Authorization is based on some type of predefined criteria, which is enforced through access control lists, security labels, capabilities tables, or user profiles. These three components of access control usually work together in a synergistic relationship and can be found in applications, operating systems, firewalls, routers, databases, domain controllers, and more.

### Role Allocation (RBAC)

The role allocation process is done with the basic RBAC model. The work of RBAC is to assign role for users and to assign permissions for that role. It regulates user access to computer resources based on different roles in the organization. This model acts as an authorization model. It allows only authorized user to access resources.

### User-Role Relationship (URR)

Single user has many roles in an organization. The administrator assigns roles to users based on the information provided in the authorization phase. Each role has different permissions. Based on the role access rights are provided to access the resources. The administrator has the responsibilities of assigning roles to the user; the user should satisfy the assignment policies. If the user satisfies the policies, the role is being assigned to the user and the compelling part of the roles framework is the ability to assign a user into multiple roles, the efficiency of each role is joined to produce the effective set of capabilities. The role can have multiple users; the role will only work if the role assignment is made in the correct background.

## Role-Permission Relationship (RPR)

Each role has one or more permissions and they are predefined for each role. These permissions are not assigned to users directly, instead it is assigned to roles. Role permissions tell which role can have access to which resources. Permissions may be as write and read access of resources. If the users have the active role, they are authorized to access the subject. Permissions can be assigned to many roles, many operations and. Roles are a collection of permissions. Users who are requiring these permissions are assigned to the selected roles. Roles availing access are selected only by the users with the administrator permission.

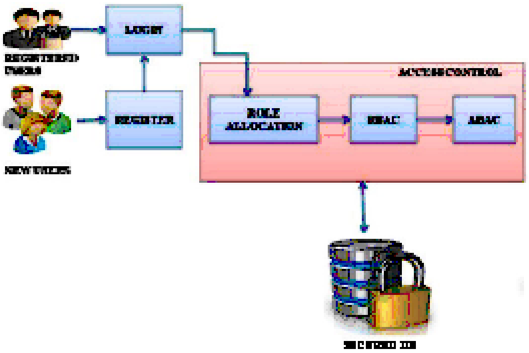

**Fig. 1.** Overview of Proposed System

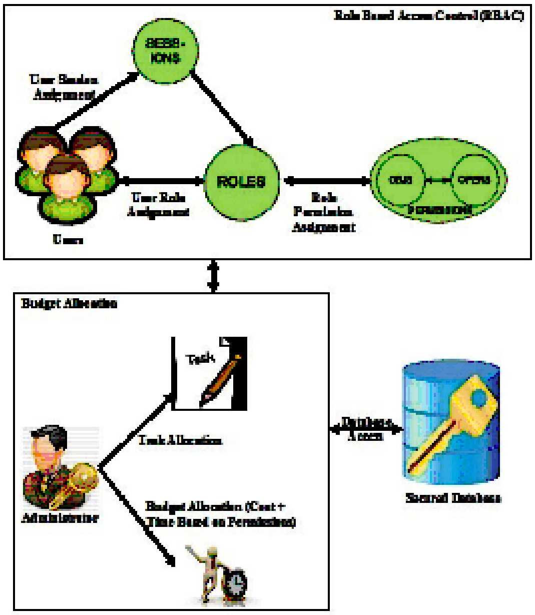

**Fig. 2.** System Architecture of Proposed Framework

Both relationships (URR and RPR) can be one-to-one or one-to-many or many-to-one relationship. Diagrammatic representation of these relations is shown in Figure 4. As we said that these proposed solutions can be implemented in university database, Figure 3 is explained by taking university roles.

For e.g., this paper considers the following,

U = {u1, u2, u3, ……… , un}
R = {r1, r2, r3, ……….. , rn}
P = {p1, p2, p3, ………. , pn}
T = {t1, t2, t3, ………... , t4}

- u1 is assigned to the student role (r1).
- u2 is also assigned to same student roles (r1).
- u3 is assigned to a teaching faculty role (r2) and counselor role (r3).
- u4 is assigned to non-teaching faculty role (r4).
- r1 has read permission (p1).
- r2 has read permission (p1) and write permission (p2).
- r3 has read permission (p1 & p2) and write permission (p2 & p3).
- r4 has read and write permission (p4).
- p1 is allowed only to read access its own personal

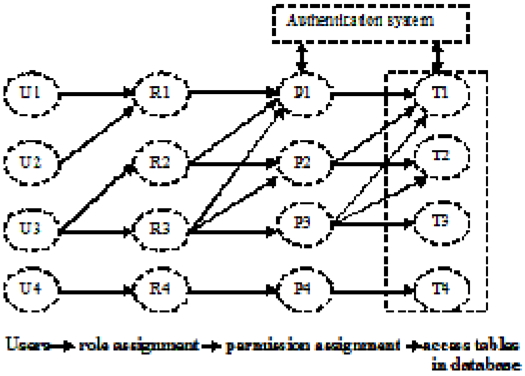
Authentication system
U1
U2
U3
U4
R1
R2
R3
R4
P1
P2
P3
P4
T1
T2
T3
T4
Users → role assignment → permission assignment → access tables in database

**Fig. 3.** Example of URR & RPR using RBAC and ABAC (U-User, R-Role, P-Permissions, T-Table)

information in student table (t1).

- p2 is allowed to read and write access its own personal information in teaching faculty table (t2) and it is allowed only to read access the personal information of students in student table.
- P3 is allowed to read access of its own personal information in teaching faculty table (t2) and allowed to write access the student's academic table and personal table (t1 & t3).
- P4 is allowed to read and write access of lab details table (t4).

These are basic work of RBAC. In our proposed work authentication system is included with access control. To access the resources permission alone not enough instead the user should cross authentication also. This is done to allow only authorized and authenticated users to access the resources.

### Attribute Based Access Control

Attribute based access control (ABAC) provides access rights and permissions based on the attributes of a user and resources. This access control provides authorization based on the attributes which is used to restrict the access. The proposed system uses ABAC to bring flexibility in URR. Here permissions are not associated directly with attributes instead permissions are associated with the role. By combining both RBAC and ABAC we can overcome the concept of static role assignment to dynamic role assignment.

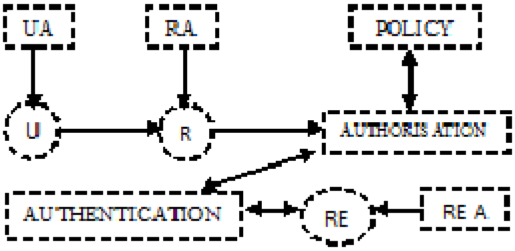

**Fig. 4.** Working of ABAC (U-user, UA-user attributes, R-role, RA-Role Attributes, RE-Resources, RE A-Resource Attributes)

Likewise, each role has some set of role attributes. This concept is called dynamic user-role and role-permission assignment. This type of authorization cannot be done by using RBAC alone. Authentication concept is also included in the proposed system. So after the permissions are provided to the user, the user should cross some authentication level also to access the resources.

### Authorization and Multilevel Authentication

#### Authorization

As it is MLSDB each user is authorized to access only particular sets of data by RBAC & ABAC access control models. These permissions are predefined and associated with different roles in an organization. By this way the system eliminates the direct disclosure of data to unauthorized and unauthenticated user. All the users use same database but views only data for which user has authorization.

#### Authentication: Multilevel Security (MLS)

An added dimension of security occurs when an information system contains resources at more than one security level. MAC can be applied in a system with different classifications resulting in Multi Level Security (MLS). Users with various security clearances are allowed to access the system concurrently, but the system only allows access to objects when a user possesses proper authorization. Therefore, if Alice has a secret clearance, then even though there may be "top secret" level information in the system, she would

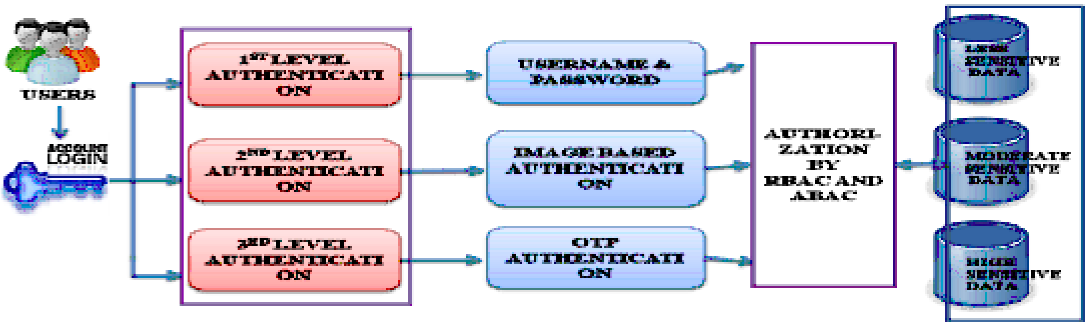
USERS
ACCOUNT LOGIN
1ST LEVEL AUTHENTICATI ON
2ND LEVEL AUTHENTICATI ON
3RD LEVEL AUTHENTICATI ON
USERNAME & PASSWORD
IMAGE BASED AUTHENTICATI ON
OTP AUTHENTICATI ON
AUTHORI ZATION BY RBAC AND ABAC
LESS SENSITIVE DATA
MODERATE SENSITIVE DATA
HIGH SENSITIVE DATA

**Fig. 5.** Authentication System

only be able to access secret and unclassified information. A benefit of an MLS system is that it alleviates the need for separate systems based on information classification. However, MLS systems are not risk-free. Physical security, inference risks, personnel security, and covert channel risks must be addressed. This paper will not deal with those issues, but they should be reviewed in the overall context of MLS systems.

Authentication is the main step to secure the resources. Authentication is interrogated whenever there is a need to establish identity. In the identity phase users need to provide some information to prove their identity. The system validates whether the provided identity matches any in the database and provide authentication. This paper implements different authentication technologies. 3-levels of authentications are implemented. The trust level of 3 levels is given as, Image based authentication > OTP > Text based password authentication.

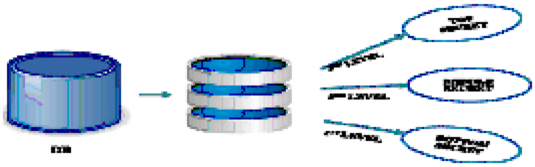

**Fig. 6.** Database Design

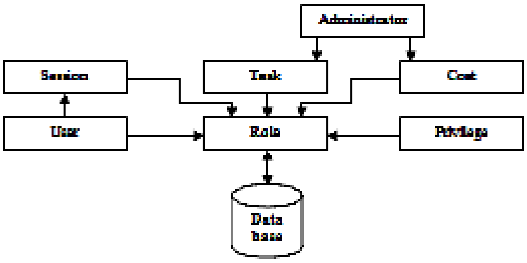

**Fig. 7.** Database Access

MLSDB can be shared by users by crossing more than one authentication level. Different roles should cross different authentication level to access the same resources.

**Level 1 - Text Based Password Authentication**

This is the basic and most used authentication technique used till date, it asks user to register first with their personal details and using their username and password as an authentication parameter we can verify user.

The registered username and password is saved in database. While logging, the user should provide the username for identification and password for authentication. If it matches with the registered one, then the system provides authentication to access the resources.

In this paper, we will store only the hash value of password, not the original password. So whenever the user log in the password is converted into a hash value and a hash is compared each time the user log in to our system. In the database the password field will be hashed, this allows maximum security, to hash the password SHA 256 algorithm is used.

In encryption, the original message from the encrypted data can be easily found using brute force method and pre-image attack or dictionary attack. Since an attacker can easily use dictionary attack and find our password, salting concept is used, where a random number is generated and this salt is used along with SHA 256 algorithm to

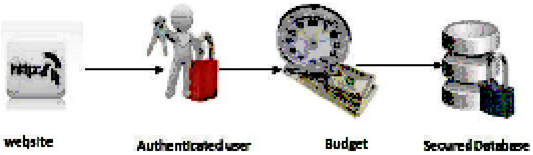

**Fig. 8.** Security to Database

**Table 1.** Comparison of Access Control Models

| Model | Constraint | Security Implication | Misuse Reduction |
|---|---|---|---|
| RBAC | Role | Effective monitoring | 75% |
| PRBAC | Privacy Policies | Detect Conflict permission assignment | 85% |
| ABAC | Attribute | Boundary well defined | 80% |
| BARBAC | Budget | Restriction to Access | 95% |
| HAC | Role, Attribute, Budget | Well Defined Boundary and Restricted Access | 98% |

generate the hash. Now this hashing is a one way function. This eliminates the risk of dictionary attack. To log in the same procedure is carried out each time. A hash is created along with registered salt and it is checked with the stored hash to authenticate users.

### Level 2 - One Time Password (OTP) Authentication

A one-time password is a set of characters that can be used to prove a subject's identity one time and one time only. After the password is used, it is destroyed and no longer acceptable for authentication. If the password were obtained by an attacker as it was being transmitted, she would have a small window of time to try and use it and most likely it was already used once, thus it is useless to the attacker. This greatly reduces the vulnerability of someone sniffing network traffic, obtaining a password, and being able to successfully authenticate as an actual legitimate user.

While registering into the organization, mail id is provided by the user. One time password is generated and it is sent to the registered mail id. This password may be random one and it is generated during the specific login session. This generated unique code is stored in a database with flag value. This unique value is used once by the user. Flag value is used to check whether the unique is used by the user. Once it is used the flag value changes and it cannot be used next time by the same user. By this way the user will be authenticated as an authentic user, and will be allowed to access the resources.

### Level 3 – Image Based Authentication

While registering for level 3 authentication the user is provided with an image in the saved imaged set. Different images are provided for different user in a random manner. In the provided image the user needs to select certain number of click points as mentioned and the user should remember the selected click points and the order of the selection. Here the click points are limited to three. The system stores those click points in X-Y pair. When the user accesses highly critical information the same image is presented to authenticate, like he/she have to click on the same area as they registered. So when a user clicks on the same area and in same pattern we authenticate them. For highly sensitive data, access level 3 is useful. According to users, they click on any object but in programming aspect the click is made on a pixel on screen, challenge is to put back the image in the same position and have to find that the user clicked point on authentication is same as on registration. By using Pythagoras theorem logic the radius of the user clicked pixel is founded and compared with the stored radius area. Here 20.0 radiuses are used. If it falls incorrect radius, then the user is not authenticated to access the resources.

### Database Design and Access

The data in the database is designed with some classification or sensitivities. As multi level authentication mechanism is implemented in the proposed work, the data in the database should be designed as highly sensitive data, moderate sensitive data and less sensitive data.

This type of database design is called as a Multi Level Security Database (MLSDB). This way of designing is used to control the access of resources.

The database can be accessed only after crossing authorization and authentication mechanism. Each user has to pass different authorization and authentication level to access the same requested resources

### Budget Allocation

The ability of user to misuse the permission while executing the task is handled by allocating the budget for each and every access right, allocation of the budget for each role is done by the administrator. Initially the total cost of a task is equal to the budget allocated to the user and the user supposed to finish their task within a given period. Let t denote the task carried out by the user, c denote the cost spend by the user for each task, you denote the user going to play a particular role, B is the total Budget. So the budget allocated for each user should be

$$B(u) = \sum_{t=1}^{n} (c_t)$$

### Task Allocation

The set of resources that are subject to access is referred by O, set of actions that can be performed on an object is referred by A and the set of all possible actions on objects is referred to as tasks T.

$$T = A \times O$$

This is the total task allocated to the user u by the administrator; it focuses on decisions by individuals about what task to perform, the role weight is being calculated by adding the cost of its associated tasks

$$r(w) = \sum_{c=1}^{n} (t_c)$$

r implies the role; w denotes the weight of the role. A role is having the logical grouping of task, users who are expected to do jobs that require some of the authority that is, the user should satisfy the particular role then only user can able to do the job.

### Cost Allocation

In Budget Aware Role Based Access Control each user should have the minimum budget to get the access permission. The user with allocated role and cost of operation can access the resource, the cost differs for different permission and it also differs for roles, administrator initially assigns cost for each permit and each role maximum budget, for example, in college the teachers may take printout 50 paise for each page and for student 1 rupee for each page depend upon the role the cost will be allocated by the administrator

$$C \rightarrow (r, t)$$

The cost C should depend upon the role r and the task t which is going to be performed, it should be managed by the administrator.

### Database Access

Earlier RBAC is fully based on the role assigned to the user, if the user satisfies the particular role they will get access permission, In our proposed model we introduce the notation called cost, cost C for executing a task t through the role r, the user should satisfy the role and budget then only they will get the access permission.

$$\text{RBAC} + \text{Budget} \rightarrow \text{Database}$$

The database should be highly secured, the administrator only can have the permission to create, delete, update the data available in the database, the data resides in the database is in the encrypted format for enhancing the security which is shown in Figure 7.

### Security in Database

Database security requires allowing or disallowing user actions on the database and data within it. Access control regulates all user access to the resources through privileges. The security will be provided in the form of authentication, authorization and auditing which is shown in Fig. 8.

**Table 2.** Budget Allocation

| Account Type | Access Time | Limit (in Rs.) |
|---|---|---|
| Current | 20 | 20000 |
| Salary | 10 | 10000 |
| Premium | 30 | 50000 |
| Saving | 15 | 15000 |

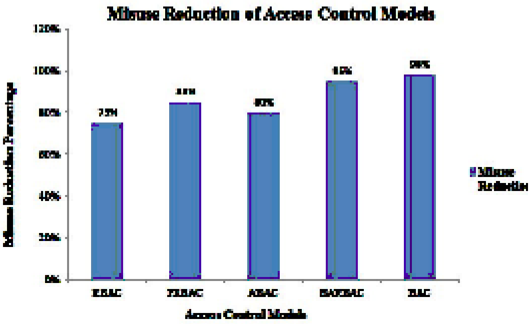

**Fig. 9.** Comparison of Access Control Models

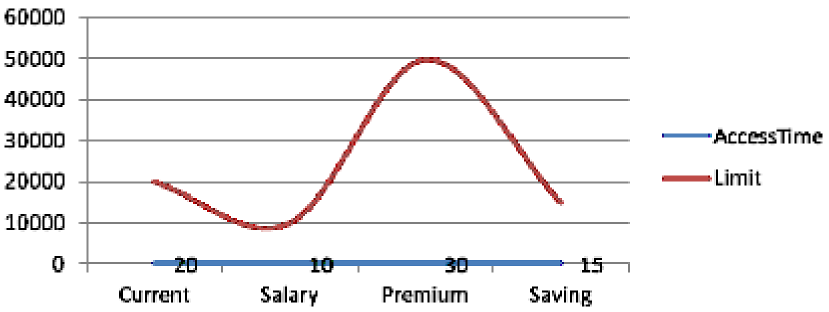

**Fig. 10.** Budget Allocation Graph

Authentication assures that the only right user gets access to the system. Authorization assures that those users only have access to resources are allowed to access. Auditing assures maintaining and monitoring when users access the protected resources. The database administrator allows a secure application to the roles that have all privileges necessary to execute the task. A role with a password can be created by the administrator to prevent unauthorized access to the privileges granted to the role.

### Delegation Implementation

It is the act of executing a task through a role that has not already been assigned to the user. This type of escalation is known as delegation. Users are authorized to delegate role r, also they are authorized to delegate a role r'. It ensures the flexibility for the RBAC. Privilege escalation occurs when a user gains more rights than were intended to be granted. In this sense, privilege means any security attribute, not just privileges. Lightening of workload is done while delegating the task to another user, while escalating the task from one to another should consider the following rules.

- The region of escalation must be defined to control the limit of escalation.
- Escalation time should be clearly defined.
- Escalation must be done by the administrator assigned way only.
- Monitor the activity of the user that is the subordinate going to perform the task and the task delegated to the subordinate, why the task was delegated those information should be maintained.
- Audit quality should be maintained for finding the cheaters.

A delegation should provide challenge for the subordinate users and encourage them to develop their efficiency. Effective delegation requires subordinate user input during the delegation process.

## RESULTS AND DISCUSSIONS

### Comparison of Access Control Models

The models are compared based on the constraints associated with each model and analyses the security implication, possibility of misuse associated with each model which is represented in Table I. RBAC is the initial access control model that is fully based on the role allocated to each user this model is used for effective monitoring. In practical RBAC has a high possibility of misuse when compared to other model which is shown in Figure 9.

Privacy aware Role based access control is based on the privacy policies assigned to each role. It could be useful in detecting the conflict between two permission assignments. Even policies are correctly specified, will have more possibility of misuse. Attribute based access control provide access to each role based on certain attribute associated with each user within an organization. Role Boundaries are well defined using this model. The proposed work is implemented to detect user misuse. It provides restriction to access the resource in the database. The user should pay the cost for each access so users may have unwillingly to misuse the provided permission.

User's misuse capability is always bounded by their allocated budget and is further adjustable through the discrimination of permission prices. Finally, it provides a uniform mechanism for the detection and prevention of misuses.

### Budget Allocation for Role

In banking framework the role is allocated based on account type constraints. Four account types are maintained, depend upon the account type role will be allocated to the customer. The budget will be provided based on two constraints that are access time and transaction limit. Access time and Transaction limit differ for each role.

The access time and the transaction limit allocated depending on the user needs represented in Table 2. The user who is having the premium account type will do more number of transaction and the person who is having salary account do less number of transaction so we allocate the limit dynamically for each role in the previous system the transaction details allocated statically to all users, when compared to static role allocation dynamic role assignment will reduce the user permission misuse and improve the dynamic nature of the system shown in Figure 10.

## CONCLUSION

This paper proposed a secured access control system which combines ABAC, RBAC and 3 level authentications. Here authorization is based either on role or attribute or both according to the request of resources made by the users. Users are allowed to access resources after crossing proper authorization and authentication step. This makes the system more secure. Existing system allows access the resources only after crossing all the three levels of authentication. So, more time is consumed to access less sensitive resources. This drawback is overcome by allowing access to resources by considering sensitivity level. By this way the proposed idea overcomes the time consumption, scalability and role assignment problems. As this concept is implemented in university database all work is done online at the university website. To access this resource we need a system with internet connection. The system is not portable to be taken to all places where we move. So it is not possible to make use of this resource at all time. In future same concept will be implemented for mobile which is portable and be carried with us anywhere. So we can make use of this resource at anytime and anywhere.

BARBAC have been implemented in the banking framework with two constraints as budget that would be useful to reduce the misuse made by the authenticated user, improve the flexibility of the working system and reduce the server processing time. The user with minimum budget can allow accessing the resources allocated by the administrator. It allows the user to delegate the task to another user; the capability of user to misuse the granted permission will be managed with budget. In Banking system delegation is implemented in the emergency situation, in that situation user send a request to the other user and the user should reply to the request then only the delegation will get complete. In future, this paper is enhanced to find a constraint for how the user should reply to the user request in case of emergency situation that will be used to increase the flexibility of the Budget Aware Role Based Access Control system.

## ACKNOWLEDGMENTS

We thank Sathyabama University for providing us the various resources and unconditional support for carrying out this work.

## REFERENCES

1. Amit Sasturkar, Ping Yang, Scott D. Stoller, C.R. Ramakrishnan, “Policy analysis of Administrative Role-Based Access Control”, *Theoretical Computer Science,* 2011; pp. 6208 – 6234.
2. Bo Lang, Ian Foster, Frank Siebenlist, Rachana Ananthakrishnan, Tim Freeman, “A flexible Attribute Based Access Control for Grid Computing”, *Journal of Grid Computing;* 2009; pp. 169 – 180.
3. Celikel E, Kantarcioglu M, Thuraisingham BM, Bertino, E, “A risk management approach to RBAC”, in Risk and decision analysis, vol. 1, IOS Press; 2009, pp. 21 – 33.
4. David F. Ferraiolo, Ravi Sandhu, Serban Gavrila, D. Richard Kuhn, Ramaswamy Chandramouli, “Proposed NIST Standard for Role-Based Access Control” *ACM Transactions on Information and System Security,* **4**: 2001, pp. 224 – 274.
5. Farzad Salim, Jason Reid, Uwe Dulleck, Ed Dawson, “Budget-aware Role Based Access Control”, Journal of Computer & Security, published by Elsevier Ltd; 2013, pp. 37 – 50.
6. Liu D, Camp LJ, Wang X, Wang L, “Using budget-based access control to manage operational risks caused by insiders”, Journal of Wireless Mobile Networks, Ubiquitous Computing and Dependable Applications, 2010; pp. 29 – 45.
7. Lorenzo Cirio, Isabel F. Cruz, and Roberto Tamassia, “A Role and Attribute Based Access Control System Using Semantic Web Technologies”, Springer-Verlag Berlin Heidelberg, 2007; pp. 1256 – 1266.
8. Ma X, Li R, Lu Z, “Role mining based on weights”, in proceeding of the 15th ACM symposium on access control models and technologies, SACMAT’10, New York, NY, USA, ACM; 2010, pp. 65 – 74.
9. Mohsen Saffarian, Qiang Tang, Willem Jonker, and Pieter Hartel, “Dynamic User-Role Assignment in Remote Access Control”, Centre for Telematics and Information Technology University of Twente, Enschede. ISSN 1381-3625 pp. 9 – 14.
10. Ni Q, Trombetta A, Bertino E, Lobo J, “Privacy-aware role based access control”, in proceedings of the 12th ACM symposium on access control models and technologies, SACMAT ’07, New York, NY, USA, ACM, 2007, pp. 41 – 50.

11. Nirmalrani V and Sakthivel P, “Design and Implementation of A-RBAC Model for Services in Distributed SOA”, in the proceedings of National Conference on Emerging Trends in Information and Communication Technologies, SRM University, Chennai, September 2012, pp. 101 – 107.
12. Patricia A. Dwyer, George D. Jelatis and Bhavani M. Thuraisingham, “Multilevel Security in Database Management Systems”, Elsevier Science Publishers, 1987, pp. 252 – 260.
13. Richard Kuhn, Edward J. Coyne, Timothy R. Weil, “Adding Attributes to Role-Based Access Control”, *IEEE Computer,* 2010; **43**(6): pp. 79 – 81.
14. Salim F, Reid J, Dulleck U, Dawson E, “An approach to access control under uncertainty”, in proceedings of the sixth International Conference on Availability, Reliability and Security, IEEE Computer Society; 2011, pp. 1 – 8.
15. Song GUO, Xiaoping LAI, “An Access Control Approach of Multi_Security Domain for Web Service”, Elsevier, *Advanced in Control Engineering and Information Science Procedia Engineering,* 2011; ; pp. 3376 – 3382.
16. Sonia Chiasson, Elizabeth Stobert, Robert Biddle, Alain Forget, Paul C. Van Oorschot, “Persuasive Cued Click-Points: Design, Implementation, and Evaluation of a Knowledge-Based Authentication Mechanism”, IEEE Transactions on Dependable and Secure Computing, *IEEE Computer Society*, 2012, **9** (2), pp. 222 – 235.
17. Surabhi Anand, Priya Jain, Nitin and Ravi Rastogi, “Security Analysis and Implementation of 3-level Security System using Image Based Authentication”, Computer Modelling and simulation (UK SIM), 14th International Conference; IEEE Computer Society; 2012; pp. 547 – 552.
18. C. Thenmozhi, S. Sathvi, B. Thamotharan, “Two Level Image Based Authentication System”, *International Journal of Engineering and Technology (IJET),* ISSN: 0975-4024, **5**(3), 2013, pp. 2036 – 2040.
19. Xin Jin, Ram Krishnan and Ravi Sandhu, “A Unified Attribute-Based Access Control Model Covering DAC, MAC and RBAC”, *IFPT International Federation for Information Processing;* 2012; pp. 41 – 55.
20. Zhao X, Johnson ME, “Access governance: ûexibility with escalation and audit”, in: HICSS; 2010, pp. 1 – 13.
21. http://www.cis.famu.edu/~hchi/langzhao_Thesis_final1.pdf.
22. http://www.slashdocs.com/nnmyvp/0072225785-ch02.html.
23. http://blogs.msdn.com/b/pepeedu/archive/2010/02/04/outlook-delegate-with-exchange-2010-rbac-implementation.aspx.