City University
...creating a culture of excellence

Proposal for

ROLE BASED USER CONTROL

FINAL PROJECT

System Analysis and Design

SUBMITTED BY (Group-1)

ABDUL KADER 171442630
MD Akbar Ali 171442628
MD Emon Shahidullah 171442627

BATCH:44TH
DEPARTMENT OF CSE

SUBMITTED TO

Supta Richard Philip
Sr. LECTURER DEPARTMENT OF CSE
CITY UNIVERSITY, BANGLADESH

DATE OF SUBMISSION: 04-07-2019

Table of Contents

Chapter: 1

Before You Begin

Before we dive into project we think it would be a good idea to explain:

- What RBUC is
- How leveraging RBUC can help you manage and improve your application's access control and security
- How Authentication and Authorization differ
- What RBUC has to offer you and your project

Why use RBUC?

- Role Based User Control is the standard means of authorization (access control). The other approach is ACLs, where a table defines who can do what. ACLs are only good for very small systems, because of the following reasons:

- Big systems have lots of permissions
- People move in organizations, and all their permissions should be changed when they do
- Maintenance (adding, changing, removing) of 100,000 permissions requires a handful of staff
- Maintenance of the permissions assigned to each user, requires more staff than above!
- One wrong user-permissions and you have a serious breach in your security, so no room for error

ACLs								
	Title	Owner Control	Promote	Modify	View Content	View Properties	Publish	Remove
	Authenticated Users				✓	✓		○
	HR Managers		✓	✓	✓	✓		○
	OS Admins	✓	✓	✓	✓	✓	✓	○
	Designers				✓	✓		○

Example of an ACL

Usage of ACLs has led to broken authorization and access control all over applications, and authorization is limited only to critical operations to keep the number of permissions low.

But RBUC is here to save the day.

What is RBUC?

RBUC separates the concepts of Users, Roles and Permissions. Roles are defined in a system, then Permissions defined separately. Then the security administrator decides what role should be permitted to do what action, by assigning that role to the permission. Finally users are assigned to roles. The system does the rest.

- Still lots of permits in the system are the problem
- People move, and only their roles need to be changed
- Maintenance of permits is still an issue
- Maintenance of permits assigned to each role is easy, it doesn't change much logically.
- Role-Permission assignments can be double checked so that no wrong permit is given to any role

RBUC requires Roles and/or Permissions to be hierarchical, so that management of them can easily be handled in hierarchies.

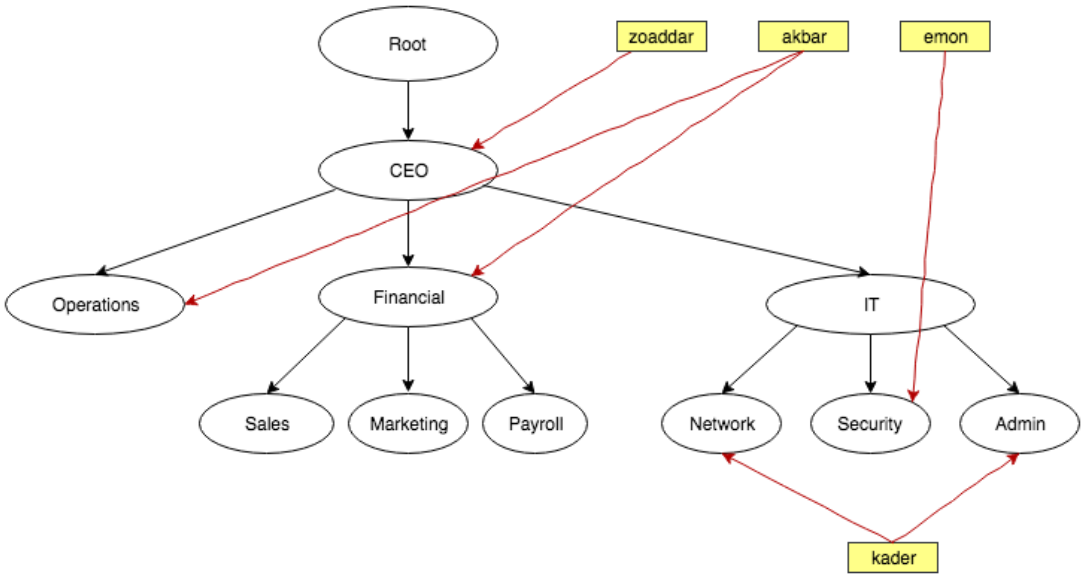
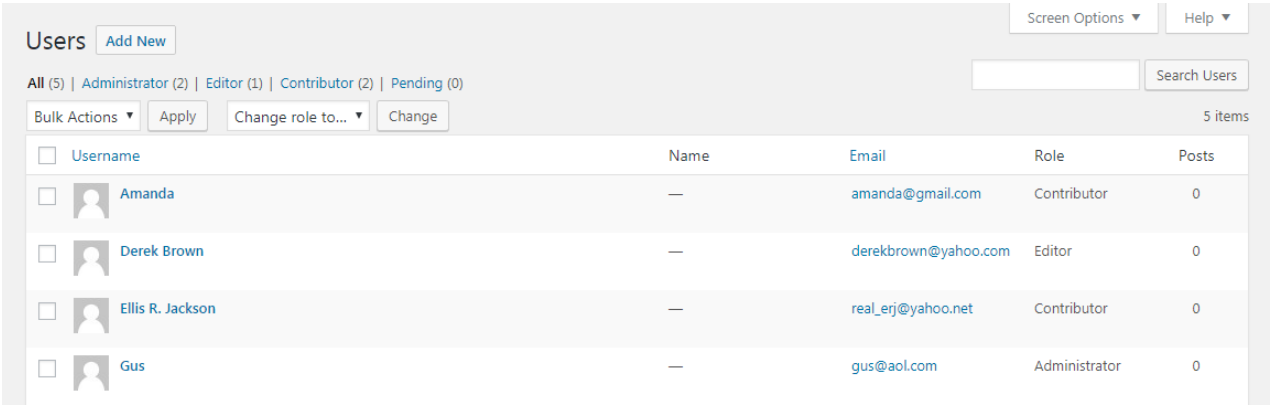


Figure:A system in hierarchical RBUC

Chapter: 2

Thefixed sixroles of WordPress user

- Administrator
- Editor
- Author
- Contributor
- Subscriber
- Super Admin



Chapter: 3

Authorization vs Authentication

A reliable Access Control System should include at least these four pieces:

- Authentication
- Authorization
- Access Approval
- Audit (audit logs/trails)

In this discussion we will focus on the two pieces that often cause confusion to those new to Access Control Systems:

- **Authentication** and **Authorization**

Authentication

Authentication... is the act of confirming the truth of an attribute of a datum or entity. This might involve confirming the identity of a person or software program, tracing the origins of an artifact, or ensuring that a product is what its packaging and labeling claims to be. Authentication often involves verifying the validity of at least one form of identification.

Authorization

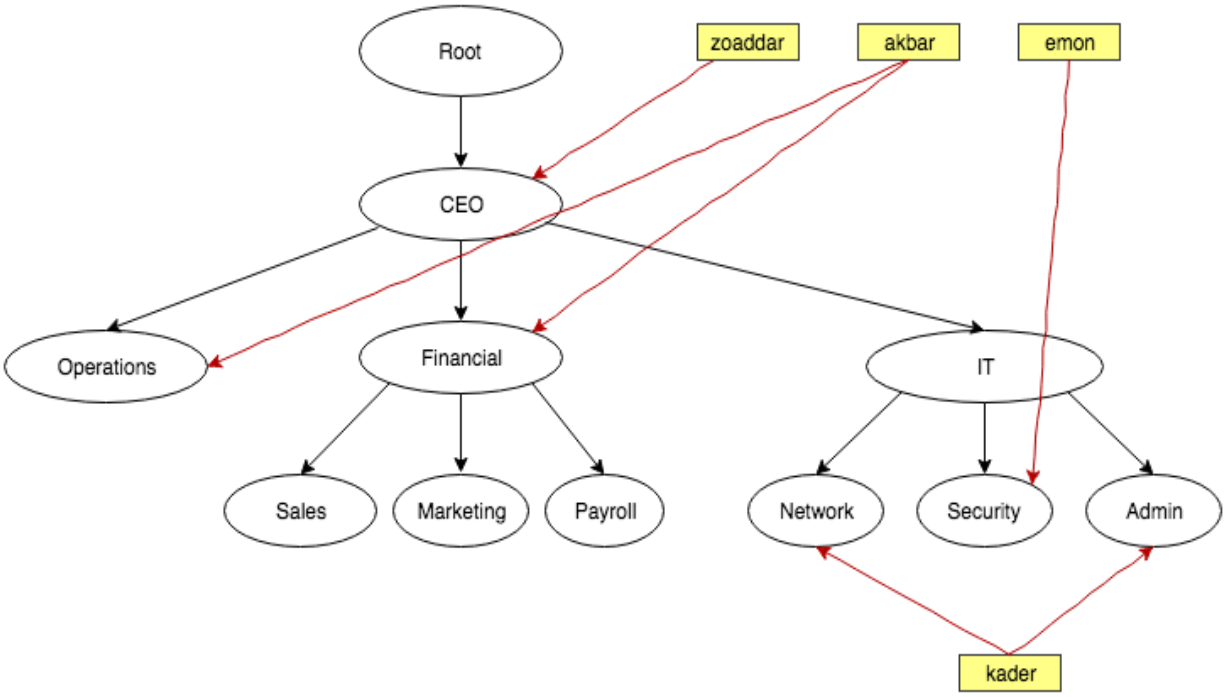
Authorization or authorization is the function of specifying access rights to resources, which is related to information security and computer security in general and to access control in particular. More formally, "to authorize" is to define access policy.

Keep in mind

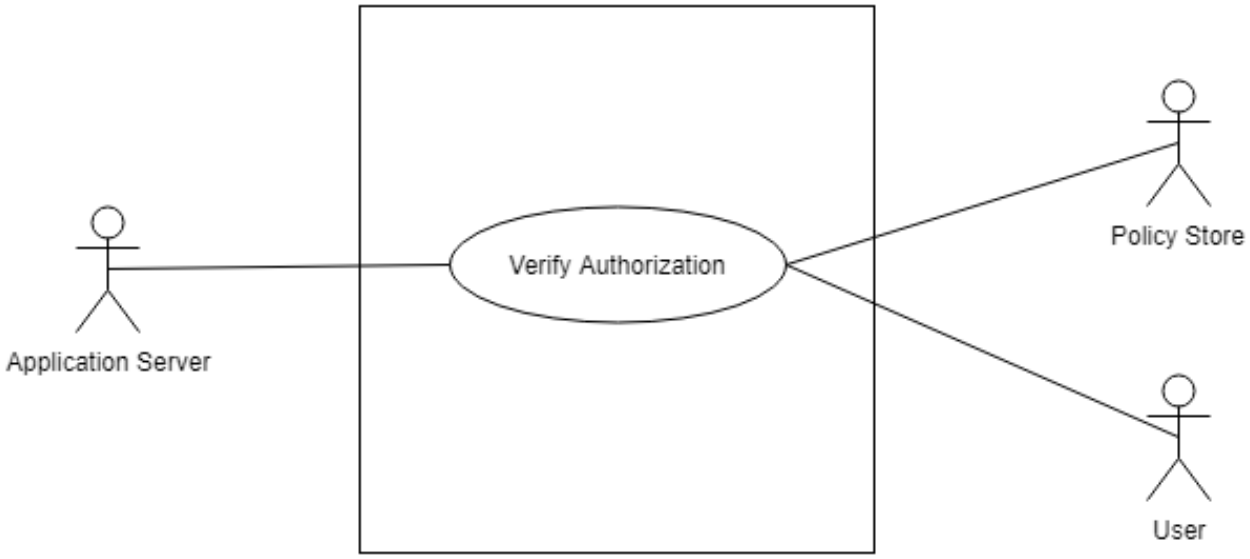
- An Authorization System allows you to easily create and manage Access Rights
- An Authorization System lets you know if an Entity has the proper Access Rights to a Resource when asked
- An Authorization System does not define application behavior (grant/deny) that is triggered when an Entity does or does not meet the requirements of a Resource's Access Rights
- It is up to your application to Authenticate your Users
- It is up to your application to grant/deny access to Resources based on what RBUC returns regarding the Users Access Rights

Chapter: 4

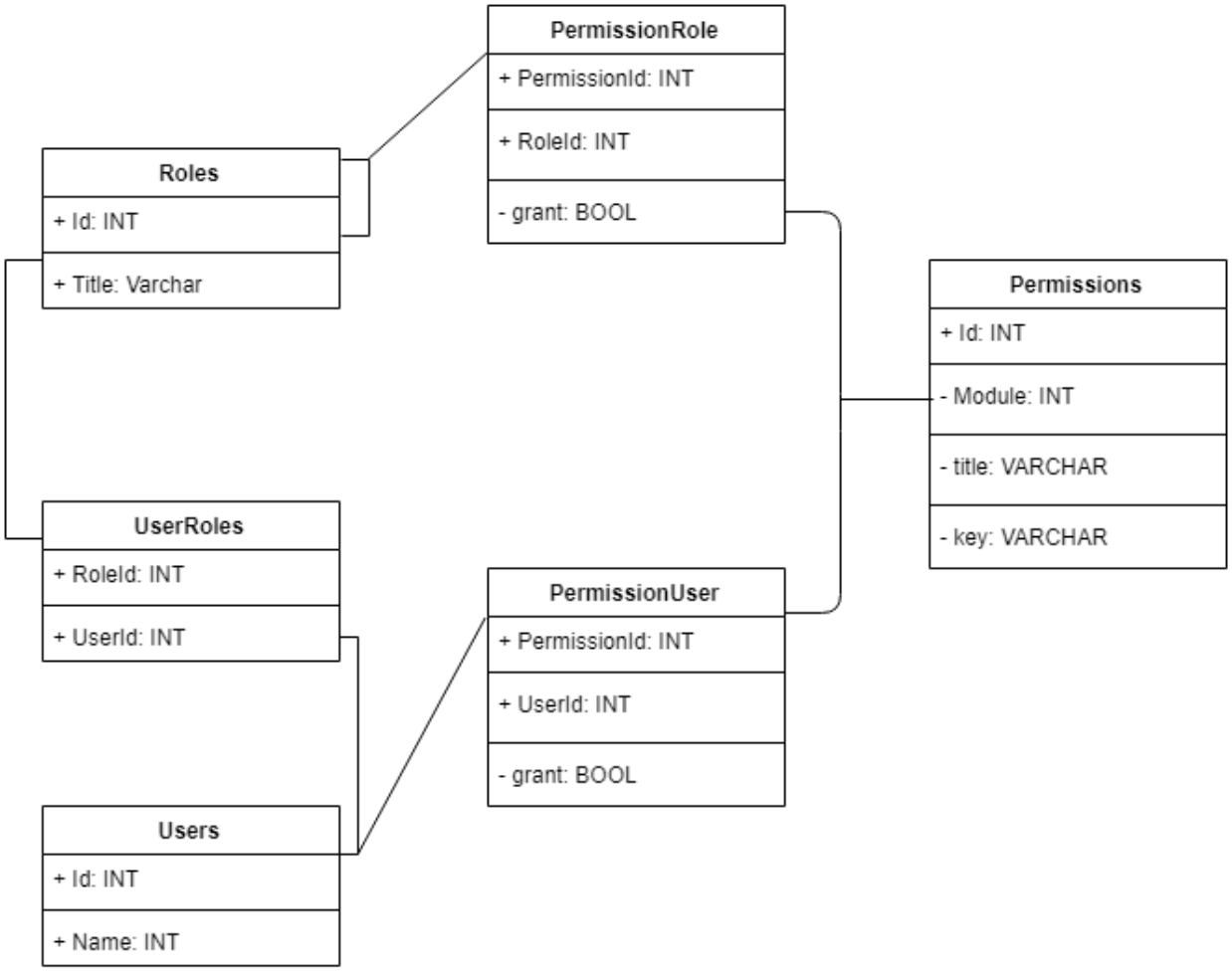
Project Structure Diagram:



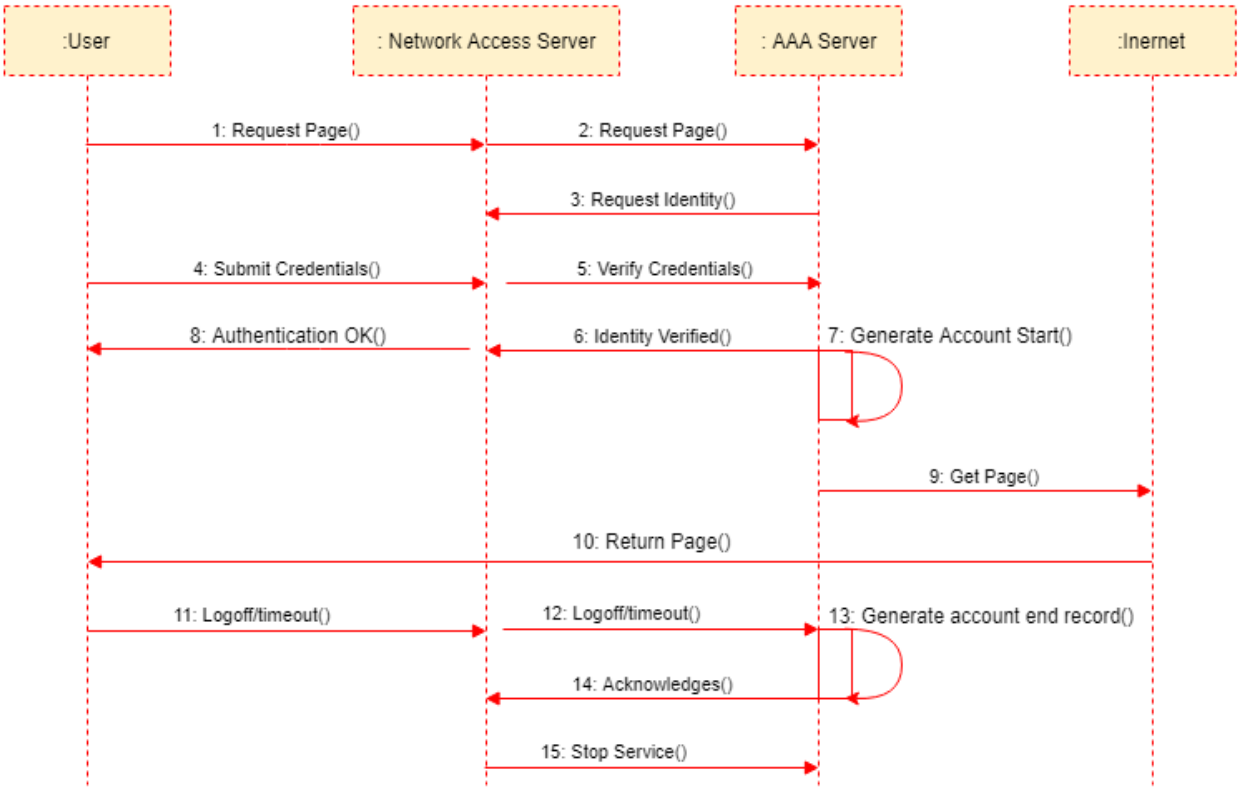
Use-case Diagram:



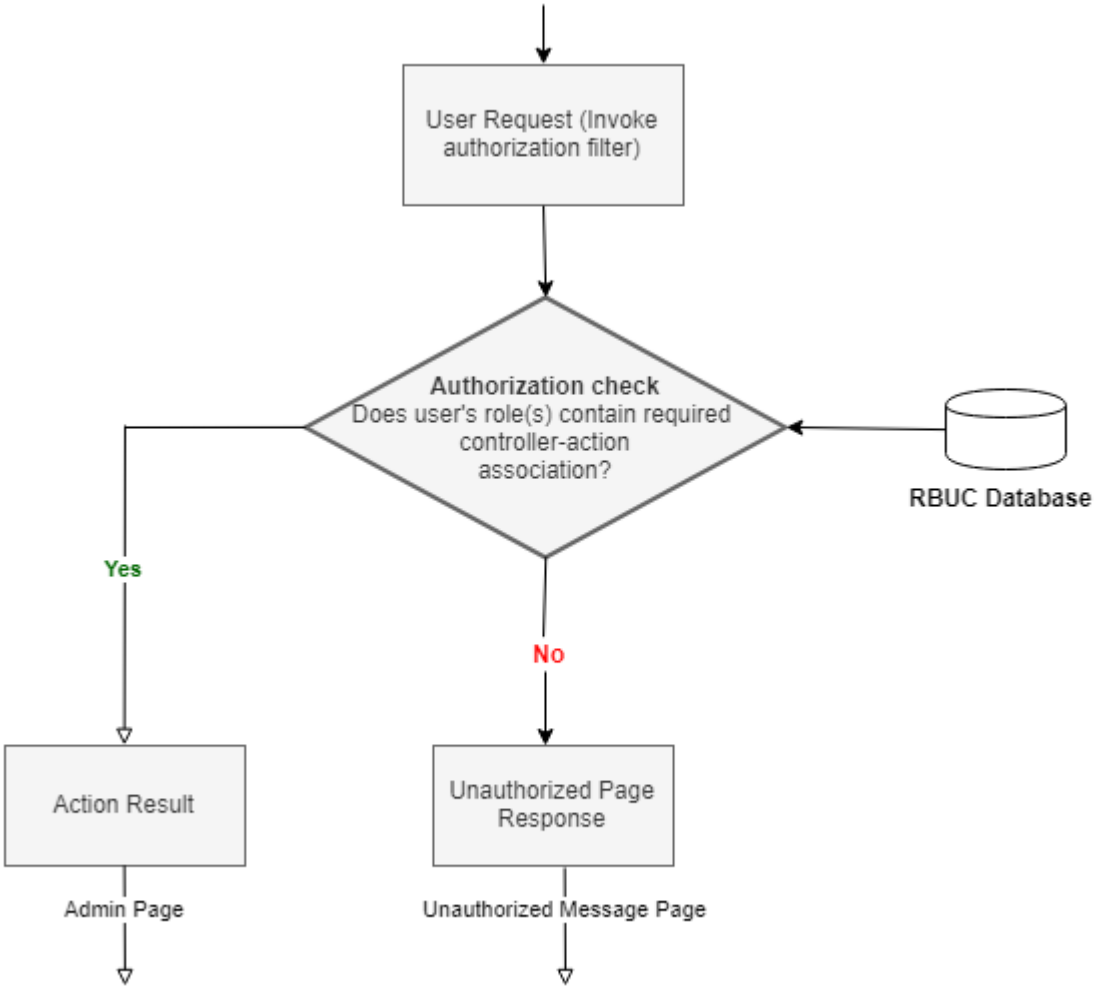
Class Diagram:



Sequence Diagram:



Activity Diagram:



Chapter: 5

What does RBUC offer?

- Standard Hierarchical Role Based Access Control
- A secure, flexible, reliable, and standardized RBUC system that will conform to your application's custom Access Control Policy needs without the headache.
- An extremely fast implementation
- Ease of use
- A sense of security

Even though we provide these tools for other developers to leverage, we use these tools in our own applications. Our goals towards quality and usefulness isn't just aimed at you. We want to make our lives easier too!