CADT

Cambodia Academy of Digital Technology

User Grant and Access

Group : SE G3

Lecturer : Mr. Thea Sophol

Team: Group 6

Members: Prak Pychey & Lim Lyheang

Topic : University Database User and Role Management

1. Workflow Overview

Our user and role management implementation followed a systematic three-phase approach to establish secure access control for the UniversityDB database.

Phase 1: User Creation Created seven users with strong passwords following naming conventions that reflect their organizational roles. Each user was configured with localhost access restrictions for enhanced security.

Phase 2: Role Definition Established seven distinct roles using MySQL's CREATE ROLE command, with each role designed to match specific job functions within the university environment.

Phase 3: Permission Assignment and Role Mapping Granted specific database privileges to each role based on functional requirements, then assigned roles to appropriate users using MySQL's GRANT statements.

2. Design Decisions

2.1 Role-Based Access Control Approach We chose RBAC over direct user privilege assignment to simplify management and improve security. This approach allows for easier maintenance and clear separation of duties.

2.2 Seven-Role Architecture Admin Role: Full database access and user management capabilities Staff Role: Read/write access to core academic records (Students, Faculty, Courses, Departments) Faculty Role: Read access to student information with write permissions for enrollments Student Role: Read-only access to relevant academic and personal information Financial Role: Specialized access to financial records and related student data Librarian Role: Complete library management with read access to user directories Readonly Role: Database-wide read permissions for reporting purposes

2.3 Principle of Least Privilege Each role receives only the minimum permissions necessary for their job functions. For example, students cannot modify any records, while faculty can only update enrollment information but not core student data.

3. Implementation Details

3.1 User Setup Seven users created: john (admin), sophia (faculty), emma (student), michael (staff), olivia (librarian), liam (finance), and noah (readonly). All passwords follow enterprise security standards with complexity requirements.

3.2 Role Permissions Admin: ALL PRIVILEGES on UniversityDB with GRANT OPTION and CREATE USER privileges Staff: SELECT, INSERT, UPDATE on Students, Faculty, Courses, Departments tables Faculty: SELECT on Students, Faculty, Courses, Departments, Library; SELECT, INSERT, UPDATE on Enrollments
Student: SELECT only on Students, Faculty, Courses, Departments, Library, Enrollments, Financial_Records Financial: SELECT, INSERT, UPDATE on Financial_Records; SELECT on Students and Enrollments Librarian: SELECT, INSERT, UPDATE, DELETE on Library; SELECT on Students and Faculty Readonly: SELECT on all tables in UniversityDB

3.3 Security Implementation Host restrictions limit all users to localhost connections. Role separation ensures no user has conflicting permissions. Only the admin role can create new users or modify permissions, preventing privilege escalation.

4. Technical Execution

The implementation used standard MySQL commands for user creation, role definition, privilege granting, and role assignment. Each step was executed systematically to ensure proper security boundaries while maintaining operational functionality for university staff and students.