PC317: DATABASE MANAGEMENT SYSTEMS
Database Security

Learning Objectives:

- Replace plaintext passwords with securely hashed passwords
- Enforce the Principle of Least Privilege by creating roles and restricted views.
- Add a trigger to an audit log table to record INSERT, UPDATE, and DELETE actions.
- Provide recommendations on how to secure data at rest and in transit.
- Fix SQL injection vulnerabilities in the provided web form using parameterized queries.

Background:

Your team is hired as security consultants for a university helpdesk system. The `helpdesk.py` application manages student tickets, staff users, and sensitive contact details. The current system's database – `helpdesk_dump.sql` is intentionally vulnerable: it has weak authentication, plaintext passwords, no role separation, no auditing, and is open to SQL injection.

Your job is to progressively secure the system across five key security layers and present your findings in a report and demonstration.

Phase 1: Authorization

Task: Replace the plaintext passwords with securely hashed passwords in the staff_users table.
Deliverables:

- A screenshot of the `staff_users` table showing the new password_hash column with hashed values.
- A screenshot demonstrating a successful login using the new, secure authentication check.

Phase 2: RBAC and Views

Task: Create specific roles (`support_role`, `admin_role`) and restricted views to enforce the Principle of Least Privilege.

Deliverables:

- A screenshot demonstrating a SELECT query on your new, restricted view that succeeds for the support_role.
- A screenshot demonstrating that a SELECT query on the raw `students` table's sensitive phone column fails with a permission error.

Phase 3: Auditing and Logging

Task: Create an `audit_log` table and a trigger to record `INSERT`, `UPDATE`, and `DELETE` actions on the `tickets` table.

Deliverable: A screenshot of your `audit_log` table showing new entries after you perform an insert, update, or delete action on a ticket.

Phase 4: Data in Transit and at Rest Recommendations

Task: This is a non-coding task. You will prepare recommendations on securing data in transit and at rest.

Deliverables:

- A brief, written explanation discussing the importance of TLS for encrypting communication between the application and the database.
- A brief, written explanation about the conceptual importance of using a Key Management Service (KMS) for protecting encryption keys, emphasizing why hardcoding keys in code is a dangerous practice.

Phase 5: SQL Injection Fix

Tasks: The provided web form has vulnerabilities to SQL injection. Fix these in the application layer using parameterized queries.

Deliverable: A screenshot demonstrating an attempted injection like ' OR '1'='1 that fails after your fix, showing an error or an empty result set instead of unauthorized data.

Final Presentation and Critique
Each group will present their deliverables and discuss their solutions. The presentation will be followed by a Q&A session where other groups will act as professional peers (e.g., DBA, Security Analyst) and critique your work. You are expected to answer the following questions during your presentation.

Discussion Questions:

1. What specific features of your password hashing function make it more secure for passwords?
2. Explain the Principle of Least Privilege in your own words. How does using roles and views directly apply this principle?
3. What steps would you take to secure the audit log table itself? Who should have access to it, and what permissions should they have?
4. In simple terms, explain how a parameterized query prevents SQL injection. What is the fundamental difference in how the database engine processes a parameterized query versus a query created

GRADING RUBRIC

Category	Points	Criteria
Phase 1: Secure Authentication	**15**	**Excellent** (15): Passwords are correctly hashed. The new authentication check is implemented and verified. **Good** (10): Hashing is implemented but with minor issues. The login check is functional. **Needs Improvement** (5): Hashing is attempted but not functional. The login check is still using the old method. **Unsatisfactory** (0-2): No attempt to implement password hashing.
Phase 2: RBAC & Views	**15**	**Excellent** (15): Roles and views are correctly implemented and functional. The Principle of Least Privilege is enforced and demonstrated with screenshots. **Good** (10): Roles and views are created, but with minor security flaws or incomplete restrictions. **Needs Improvement** (5): Roles and views are attempted but are not correctly enforcing access controls. **Unsatisfactory** (0-2): No attempt to create roles or views.
Phase 3: Auditing & Logging	**15**	**Excellent** (15): The audit log table and trigger are correctly implemented and capture all required actions (INSERT, UPDATE, DELETE). **Good** (10): The audit log is functional but may have minor issues (e.g., missing data points or actions). **Needs Improvement** (5): The audit log is created but the trigger is not functional. **Unsatisfactory** (0-2): No attempt to implement auditing.
Phase 4: Recommendations	**10**	**Excellent** (10): Recommendations on TLS and KMS are clear, well-written, and demonstrate a strong understanding of the concepts. **Good** (7): Recommendations are provided but may be generic or lack specific details. **Needs Improvement** (4): Recommendations are minimal or show a basic, but not thorough, understanding. **Unsatisfactory** (0-2): No recommendations provided.
Phase 5: SQL Injection Fix	**15**	**Excellent** (15): The SQL injection fix is correctly implemented using parameterized queries. The vulnerability is proven to be fixed via screenshots. **Good** (10): The fix is implemented but may have minor issues (e.g., incomplete fix, minor syntax error). **Needs Improvement** (5): A fix is attempted but is not a parameterized query. The vulnerability is still present. **Unsatisfactory** (0-2): No attempt to fix the vulnerability.

Final Presentation	**15**	**Excellent** (15): Group presents clearly, professionally, and within the time limit. All required deliverables are shown and explained. **Good** (10): Presentation is clear but may be disorganized or exceed the time limit. **Needs Improvement** (4): Presentation is disorganized and difficult to follow. **Unsatisfactory** (0-5): No presentation given.
Q&A and Critique	**15**	**Excellent** (15): Group members provide clear, insightful, and correct answers to all discussion questions. They show a deep understanding of the concepts. **Good** (10): Group members answer most questions correctly but may struggle with more complex concepts. **Needs Improvement** (5): Group members struggle to answer the questions or provide incorrect information. **Unsatisfactory** (0-2): No answers provided.