CS541 - Project

Authors: Amber Johnson, Craig West
18 April 2014

A. **Task Breakdown**
B. **Roles of Group Members**
C. **Design of the Database and Application**
D. **How the Design Works**
E. **Additional Features**

A. Task Breakdown

The ER and Model diagrams were created for visual guidance for creating the application. Next, the data was generated for the relations. The application structure was defined corresponding to each policy of the Biba Integrity Model.

B. Roles of Group Members

The diagrams were created by Craig West while the constructing of the relations and data was completed by Amber Johnson. The application structure and implementation was a collaborative effort between all group members.

C. Design of the Database and Application

i. Design of the Database

The design of the database can be found in the attached *Model-Diagram and ER-Diagram* files.

ii. Design of the Application

To demonstrate the Biba Integrity Model, users are allowed to login either as students or teachers, depending on their status. The users are allowed to perform selections, inserts, and updates which correlate to reading and writing in the Biba Model. In order to demonstrate different modes there is a variable that can be set which determines which mode the application runs in. The

application runs in an infinite loop allowing users to perform as many queries as they would like. We restrict the queries one can perform by explicitly asking the user which query to execute, and prompt the user for input as a guideline. When the queries are being executed they are sanitized in case any bad data is inserted, preventing malicious usage.

D. How the Design Works

Upon running the application, data is inserted into the database (provided the person running the application inserts the proper credentials for connecting). The user is prompted to log in as a teacher or student. The user is then asked to specify their action by entering the coordinating numerical value (Ex. 1. Select, 2. Insert). Next they are prompted for the relation in which they wish to perform the action and asked to enter a predicate. Integrity values are stored in a table *Integrity*, which stores the integrity values of relations and users and their names and id values as the primary key.

Biba Model Implementation

i. Strict Policy

When running the application in the strict policy mode, the user's integrity level is extracted and stored into a variable upon login. If the user tries to perform an update or insert on a table, a method is called to compare the integrity value of the user and the table that they are attempting to access. If the designated table has a higher integrity value than that of the user, an error will be thrown and the program will loop back to the main menu, satisfying the simple integrity axiom or "no write up". Like the simple integrity axiom, the application also satisfies the star integrity axiom or "no read down" by not allowing users to perform a select operation on a relation with a lower integrity level than their own.

ii. Object Low-Watermark

Entering the "Watermark" mode of the application allows users to perform *Updates* and *Inserts* on any relation regardless of the integrity levels, however, if a user (subject) performs one of these actions, the integrity level of the relation (object) will be lowered to that of the user's. Also, users are allowed to perform *Select* operations on relations if the integrity level of the relation is higher than that of the user. This satisfies the object -low watermark policy.

iii. Ring Policy

The ring policy mode in the application allows users to perform *Select* operations on all relations, but prohibits *Updates* and *Inserts* on relations with lower integrity than themselves.

E. Additional Features

i. Passwords are stored using a SHA-256 cryptographic hash function.
ii. Prepared statements and stored procedures are used to sanitize data input and prevent sql injections.