THE UNIVERSITY OF NEWCASTLE AUSTRALIA

SCHOOL OF INFORMATION AND PHYSICAL SCIENCES

School of Information and Physical Sciences

SENG2250 System and Network Security

Assignment 3

This assignment is to be done individually.

*Due on **Sunday, 22 October, 11:59pm**, electronically via the "Assignment 3" submission link in Canvas.*

Total 100 marks

***Summary**:* For this assignment, you will modify a http server program to provide authentication and access control for its various services.

***Problem statement**:* An emerging start-up, Mako, has had some recent success and so have started developing an online portal for their expanding number of employees. Having seen the failings of many other businesses in recent times, the founder of the company has decided that security is of utmost importance for their online portal. This has led them to employ you, an emerging cyber security expert, to add authentication and access control services to their online portal in its early phases of development.

***Requirements**:* You will use the either the Java or Python skeleton application[1] provided with this assessment and add the following features:

- **Admin console**. This console can only be accessed by clients that are authenticated as a user in the admin group. It includes a collection of functions that involve modifying other clients, namely add user, modify user, and delete user. Add user allows the admin to add a user to the system. The admin specifies a username and email, and the server generates a random password for that user and sends it to them along with the username in an email. Modify user allows the admin to change the group that a user belongs to. Delete user allows the admin to remove the specified user. **(20 Marks)**
    - The program should start up with one default user, root, who is a part of the admin group, and starts with a randomly generated password which is printed by the server program.
- **Password storage**. The stored passwords of users in the system should follow good practices. **(10 Marks)**
- **Multi-factor authentication**. Before requesting any service, the client sends the server their username and password, the server then sends a code to the client's email, which the client must reply with. To handle the sending of the email, we recommend using:

1. You may use a language other than Java or Python, just note that you will likely have to translate the skeleton programs.

https://www.mailgun.com/. **(20 Marks)**

- **Token authentication**. After multi-factor authentication, the client should be given a "token", which is valid for 15 minutes from being issued. As long as the client provides a correct and valid token to the server, the server will not require multi-factor authentication. The token itself should be a unique and hard to guess/predict value (that is a string or number). **(20 Marks)**
- **Access control**. There are several services implemented in the server skeleton program, you will add the Biba access control model to them, where object labels of each service are given in Table 1. All clients are assigned "rw" permissions for all resources at the access control matrix level, so their access is purely determined according to the rules of the Biba model. Additionally, we provide the read and write endpoints in Table 1, these correspond to the read/write function in the server code for that row's resource. **(16 Marks)**
    - This access control model does not apply to the admin console, it instead can only be accessed (read and write) by users in the admin group.
    - The admin group also has the security level of Top Secret.
    - In total, we have three security levels: Top Secret, Secret, and Unclassified. Their hierarchy is in that same order Top Secret > Secret > Unclassified.

Resource	**Security Level**	**Read Endpoint**	**Write Endpoint**
Finances	Top Secret	audit_expenses	add_expense
Timesheet	Top Secret	audit_timesheets	submit_timesheet
Meetings	Secret	view_meeting_minutes	add_meeting_minutes
Roster	Unclassified	view_roster	roster_shift

Table 1: Object Labels for the Online Portal Resources

- You will also write a small client program to demonstrate these features, it is sufficient to test and demonstrate each one simply procedurally. **(4 Marks)**
- Further details specific to the Java or Python programs are given in the README.md file that is in their respective folders in the starter programs available on Canvas.

Reflections **(10 Marks)**

As a capstone this assessment, you will write a brief reflection on the program you have helped to develop in the first part of the assignment. This reflection should be about 600-1000 words of length and will firstly discuss what you have learnt from extra resources, such as websites, textbooks, or large language models, to complete the tasks. The reflection will then relate what you have learnt to subjects covered in our labs and lectures. To strengthen your reflection, you will also discuss the limitations of

your developed program, such as scalability or the limited environments where it may be securely used, for each of those limitations introduce a technology that would likely address it (e.g. SSL, IPSec, etc.).

Your reflection should cite the extra resources you learnt from, if any, including prompts from LLMs (such as ChatGPT). Your bibliography should follow a citation standard such as IEEE, APA, or Harvard. The bibliography will not count towards your word count.

Submission Guidelines

Please submit your work in a single zip file. This file should encompass a PDF document illustrating program execution with running examples (including screenshots to validate the successful implementation of requirements) and reflections. Additionally, include a folder named "src" that contains all of your code and other code adjacent files required to run your programs, it also will contain a "README.md" file which states instructions for setting up and running your programs.

Marking Rubric

Requirement	Basic	Sound	Good	Excellent
Admin Console	(5) Only admin users can access the admin console	(10) The console can be used to add new clients	(15) The console can be used to delete existing clients	(20) The console can be used to modify the group of existing clients
Password Storage	(2) The server stores client passwords	(5) Passwords are stored in a way that the server cannot clearly read them	(8) The server can check that a client provided password matches the stored one	(10) The client provided password is sent to the server in a way that it still be matched to the stored one, but cannot be clearly read either
Multi-factor authentication	(5) The server stops unauthenticated clients from accessing any service	(10) When the server receives a correct username and password, they generate a unique code	(15) The server sends the unique code to that user's email	(20) The client can use that code to access the server (or gain a token if you have working token authentication)
Token Authentication	(5) The server provides the client a token	(10) The token is a unique and hard to guess value	(15) The token can be used for authentication	(20) The token remains valid for only 15 minutes
Access Control	(4) Access control mechanisms are only applied to one of the resources	(8) Access control mechanisms function correctly for two resources	(12) Access control mechanisms function correctly for three resources	(16) Access control mechanisms function correctly for all four resources
Client program	(1) The client program tests all the endpoints as single user	(2) The client program tests all endpoints as each type of user (differing groups/labels)	(3) The client program uses a token (or imitation, if there is no token authentication) where possible	(4) The client program tests each possible unique response that the server can give (token, no token, different labels, groups, etc.)
Reflections	(2) Only includes a list of what was learnt from external resources	(5) Includes a discussion of what was learnt from external resources	(8) Along with the discussion, includes a reflection of how the task relates to our lecture and lab content	(10) Along with the discussion and lab + lecture reflection, discusses the limitations of the developed program, and potential ways of addressing them

**Note: Marks may fall in between these cases for submissions of intermediate qualities, or if a lower-level feature is missing despite there being a higher level one.