CS 370 Introduction to Security **Week 5: Problem Set 5**
Instructor Name: Rakesh Bobba

Introduction

The purpose of this assignment is to help you gain a better understanding and insight into access control concepts covered in Week 5.

Before beginning make sure you have watched the lecture videos on the following and completed the associated practice quizzes.

- Introducing Access Control
- Access Control Matrix: An Abstraction
- Changing Access Policy
- Discretionary Access Control in Practice

Please make sure you read Chapter 4 of the textbook, up to section 4.2.7

Questions

Please answer the questions below.

Access Control Concepts

Q1[6 pts]: State and define the three most important components in access control, all starting with the letter 'A'?

- Authorization: Granting a right/permission to the system entity to access a system resource.
- Authentication: Binding an external entity to a system entity.
- Audit: Independent review of system actions.

Q2 [4 pts]: What is the primary difference between DAC and MAC access model?

- In DAC, regular users can adjust policy, whereas in MAC, regular users can not adjust policy.

Q3 [4pts]: In access control, what does an "open policy" and "closed policy" mean?

- Open policy: access rights default to allowing everyone access unless they are blacklisted.
- Closed policy: access rights default to not allowing anyone access, unless they are whitelisted.

Q4 [4 pts]: Explain the difference between Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC)

- Role-based: access policy is defined in terms of roles, not individuals.
- Attribute-based: Access policy is defined in terms of user, resource, and environment/context attributes.

Access Control Matrix

Consider the following scenario. An organization employs product managers, programmers and testers. The organization operates with the following kinds of files: development code and executables, testing code and executables, test reports, and production code and executables.

Product Managers can view and execute the development executables and production executables to verify correctness. Programmers can create, edit, delete, and execute development code and executables.

Programmers can also promote development code to the test level.

Testers can edit, delete, and execute test code and executables. The testers write test reports that can be read by everyone. The testers can promote test code to production level or demote it back to development.

Everyone can view and execute production code and executables.

Eve is the product manager, Alice and Bob are programmers. Carol and Dave are testers

Q5 [3 pts]: Define the rights the access control system would need to enforce the requirements for this scenario. Associate an abbreviation that you can use in the following questions.

- View: R
- Execute: X
- Write: W
- Create: C
- Delete: D
- Promote: P
- Demote: M

Q6 [7 pts]: Design an access control matrix for the scenario above for the users mentioned.

	Dev code/exes	Test code/exes	Test reports	Prod. Code/exes
Eve	R, X		R	R, X
Alice	C, W, D, X, P		R	R, X
Bob	C, W, D, X, P		R	R, X
Carol		W, D, X, P, M	W, R	R, X
Dave		W, D, X, P, M	W, R	R, X

Q7 [3 pts]: Assume the Access Matrix is being implemented by a system using Access Control Lists. Write the Access Control List for the Development Executables.

	Eve	Alice	Bob	Carol	Dave
Dev Code/exes	R, X	C, W, D, X, P	C, W, D, X, P		

Q8 [3 pts]: Assume the Access Matrix is being implemented by a Capability system. Write the Capability list for Alice.

	Dev code/exes	Test code/exes	Test reports	Prod. Code/exes
Alice	C, W, D, X, P		R	R, X

Changing Access Control Policy/Matrix

	File 1	File 2	File 3	File 4	Subject A	Subject B	Subject C
Subject A	Own R W		Own R W		Control		Own
Subject B	R	Own R W	W	R*		Control	
Subject C	R W	R		Own R W			Control

Q9 [4 pts]: Keeping in mind the rules governing access control matrix change covered in class, and the access matrix shown above, answer whether or not the following changes to access matrix are allowed. **Explain in one sentence why or why not**.

a) (allowed / not allowed) Subject C wants to Transfer R on File 2 to Subject A
- This is not allowed because Subject C does not own File 2, so cannot change permissions for other users.

b) (allowed / not allowed) Subject A wants to Delete R on File 2 from Subject C
- This is allowed because Subject A owns subject C, so can change Subject C's permissions at will.

UNIX Permissions

Q10 [5 pts]: When a file in Unix is protected with mode "644" and is inside a directory with mode "730" can you describe a way in which the file can be compromised?

- The file can be compromised when either the owner or someone within the group with permissions to the file modifies the directory. In order to modify the file, someone just needs Write permission on the directory, so the owner or someone in the group can modify the file name or even delete it, since the owner's permissions are Read, Write, and Execute, while the group's permissions are Write and Execute.

Q11 [2 pts]: Suppose you are working as the security administrator at xyz.com. You set permissions on a file object in a network operating system which uses DAC (Discretionary Access
Control). The Extended ACL (Access Control List) of the file is as follows:
Owner: Read, Write, Execute

User C: Read, Write, -
User B: Read, Write, Execute
Sales: Read, -, -
Marketing: -, Write, -
Mask: Read, Write, -
Other: Read, Write, -

User "A" is the owner of the file. User "B" is a member of the Sales group. What effective permissions does User "B" have on the file?

- Read access only

Submission Details

Submit a PDF file with the questions and your corresponding answers
The assignment is worth 45 points. It is due Wednesday of Week 6 at Midnight.