**Hsiang Lo**

**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’?

The triple As deals with access control part of a broader context. Authentication, which bind external entity to system entity. Authorization, which grant a right or permission to the system entity to access a system resource. And lastly, audit, which interpedently reviews system actions.

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

DAC or Discretionary Access Control focuses on decisions made based on dentity and access rules and regular users can also adjust the policy. While MAC or Mandatory Access Control focuses on decision made by testing labels associated with processes and resources against system policy rules. Also the regular users cannot adjust the policy.

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

In access control, open policy implies access limitations are specified while all others allowed. In closed policy, access limited to those explicitly stated.

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

RBAC or Role-Based Access Control focuses on access policies defined in term of roles rather than individual requestors. While attribute-based access control focuses on access policies defined in terms of attributes of the user, resource and environment or context for the access.

## 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.

The rights the access control system would need are create, edit, delete, execute, promote, view, demote and write.

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

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Dev Code | Dev  Exe | Test  Code | Test  Exe | Product  Code | Product  Exe | Test  Report |
| Eve (PM) |  | View, Execute |  |  |  | View, Execute | read |
| Alice (program) | Create, edit, delete, execute, promote | Create, edit, delete, execute |  |  |  | View, execute | read |
| Bob (program) | Create, edit, delete, execute, promote | Create, edit, delete, execute |  |  |  | View, execute | read |
| Carol (test) |  |  | Promote,  Demote, edit, delete, execute | Edit, delete, execute |  | View, execute | Write, read |
| Dave (test) |  |  | Promote,  demote | Edit, delete, execute |  | View, execute | Write, read |

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.

Using ACL, the development executables are

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Eve | Alice | Bob |  |  |
| Dev Execute | View, Execute | Execute, create, edit, delete | Execute, create, edit, delete |  |  |

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Dev Code | Dev Exe | Product Exe | Test Report |
| Alice | Execute, create, edit, delete, Promote | Execute, create, edit, delete | View, Execute | read |

## 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**.

1. (allowed / not allowed) Subject C wants to Transfer R on File 2 to Subject A

Yes because subject C have access to R file 2, and can have access to transfer ability to transfer R.

1. (allowed / not allowed) Subject A wants to Delete R on File 2 from Subject C

Not allowed. Subject A does not own file 2, therefore do not have access to delete

## 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?

When a file in Unix is protected with mode “644” and is inside a directory with mode “730”, this can be further analyzed by reading 644 as read write for owner, read for group and read for others. 730 refers to read write and execute, write and execute for group and no permission for others.

Looking at this, it can be seen that despite the file being none executable, the directory is allowing execution. What’s more, the directory also allows write as well, which could corrupt the file that is intended to be read only. This is bad design I believe that could lead to many more problems.

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?

Because we are looking at DAC or Discretionary Access, regular people can give access to others. Since User A is the owner of the file, user A have access to read write and execute. While User B is the member of the sales group, who also have read write and execute. But because he is also a member of the marking them, who only have access to read, user b has access to individual read write and execute, group only have write. SO I would say he can still read write and execute.

# 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.