M.Bishop: Chapter 6(pg-75) question 1, chapter 5(pg-62) question 2,6, chapter 2(pg-35) question 1,chapter 4(pg-59) question 3,4,5
William Stallings: Chapter 13(pg-482) Review question 1-5, Problem 1-5 (check once before solving,some may not be required)
COMPLETE

Foundations of Information Security by Andress, Jason, chapter 3 (pg -95)

Q1. Discuss the difference between authorization and access control.

Ans:

Aspect	Authorization	Access Control
Definition	Process of determining permissible actions for authenticated parties	Mechanisms and policies to enforce permissions for accessing resources
Purpose	Defines scope of permissible actions based on user identity and role	Regulates access to resources to ensure confidentiality, integrity, and availability
Focus	Specifies permissions granted to authenticated parties based on roles or attributes	Implements policies/rules to enforce permissions and manage access to resources
Relationship	Follows successful authentication process, validating user identity and attributes	Enforces authorization decisions made during authentication
Actions	Grants or denies permissions to perform specific actions or operations	Controls access to resources based on permissions assigned to users or groups
Components	Policies, rules, permissions, user attributes, roles	Mechanisms, tools, systems, access control lists (ACLs), encryption algorithms
Examples	Role-based access control (RBAC), attribute-based access control (ABAC), discretionary access control (DAC)	Authentication mechanisms (passwords, biometrics), encryption, firewalls, intrusion detection systems (IDS)

Q4. Which should take place first, authorization or authentication?

Ans:

Based on information security principles, authentication should take place before authorization. Authentication is the process of verifying the identity of a user or entity, ensuring that they are who they claim to be. Once authentication is successfully completed, authorization can then occur, where the system determines the specific actions or resources that the authenticated user or entity is allowed to access based on

their identity, role, or attributes. Therefore, authentication logically precedes authorization to ensure that only legitimate users gain access to the system before permissions are granted or denied.

Q5. What are the differences between the MAC and DAC models of access control?

Ans:

Feature	Mandatory Access Control (MAC)	Discretionary Access Control (DAC)
Authority to Set Access	Central authority or system administrator sets access controls for resources.	Resource owner has the authority to determine access permissions.
Enforcement Mechanism	Enforces access controls based on security labels or sensitivity levels associated with resources and users' security attributes.	Relies on the discretion of resource owners to grant access permissions based on user identities and group memberships.
Applicability	Predominantly used in highly secure environments, such as government and military organizations, where strict access controls are necessary to protect sensitive information.	Commonly found in everyday computing environments like operating systems and file systems, allowing resource owners to control access to their data.
Flexibility vs. Rigidity	Offers a rigid and centrally controlled approach to access control, prioritizing security over flexibility. Access decisions are based on predefined security policies and classifications.	Provides flexibility as resource owners can tailor access permissions according to their preferences and requirements.
Access Decision Criteria	Access decisions are typically based on security clearances, sensitivity labels, and predefined security policies. The principle of least privilege is often applied to grant minimal necessary access.	Access permissions are determined based on the discretion of resource owners, considering factors such as user identities, group memberships, and permissions assigned to objects.
Examples	Commonly implemented in environments with strict security requirements, such as classified government networks and highly regulated industries.	Frequently used in personal computing environments, network file systems, and collaborative platforms where users have control over their data.

Q6. The Bell–LaPadula and Biba multilevel access control models both have a primary security focus. Can these two models be used together?

Ans:

Yes, the Bell–LaPadula (BLP) and Biba multilevel access control models can be used together, although they have different primary security focuses. BLP primarily focuses on maintaining confidentiality by restricting unauthorized access to sensitive information, while Biba prioritizes data integrity to prevent unauthorized modifications.

Despite their different focuses, these models can complement each other when implemented together in certain environments where both confidentiality and integrity are critical. By combining the principles of both models, organizations can achieve a more comprehensive security posture that addresses both aspects of information security: confidentiality and integrity.

In practice, the BLP model can be used to enforce restrictions on reading sensitive data based on security clearances, ensuring that users cannot access information above their authorized clearance level (no read up). On the other hand, the Biba model can prevent unauthorized modifications to data by enforcing restrictions on writing data to resources classified at lower integrity levels (no write down).

By integrating these two models, organizations can establish a robust security framework that safeguards against unauthorized access and tampering of sensitive information, thereby enhancing overall data protection and compliance with security policies.

Q10. What are some of the differences between access control lists and Capabilities?

Ans:

Aspect	Access Control Lists (ACLs)	Capabilities
Definition	Lists attached to resources specifying who can access them and their permissions	Tokens or keys granted to processes specifying the resources they can access
Granularity	Typically defined at the level of resources (e.g., files, directories)	Typically defined at the level of processes or threads
Flexibility	Provides flexibility in managing permissions for multiple users or groups	Offers more fine-grained control over access by directly associating capabilities with processes
Dynamicity	Changes to permissions usually require modifying the ACLs associated with resources	Capabilities can be dynamically created, revoked, or transferred during runtime
Management complexity	May become complex to manage when dealing with large numbers of users or resources	Generally simpler to manage as capabilities are directly associated with processes
Revocation of access	Revoking access for a user often involves modifying ACLs for each affected resource	Revoking capabilities from a process is straightforward and affects only that process
Network environments	Commonly used in network-based access control mechanisms, such as firewalls	Less common in network environments due to challenges in distributing and managing capabilities across distributed systems
Dependency on object state	Access permissions are dependent on the state of the resource (e.g., file permissions)	Capabilities are independent of the state of the resource and are granted based on the authority of the process

Information-security-principles-and-practice, Chapter 8 (pg-328)

16. Combine BLP and Biba's Model into a single MLS security model that covers both confidentiality and integrity.

Ans:

Combining the Bell–LaPadula (BLP) and Biba models into a single Multilevel Security (MLS) model requires careful consideration of how to address both confidentiality and integrity requirements simultaneously. Here's an approach to integrate elements of both models into a unified MLS framework:

1. **Hierarchical Classification Levels**: Define a hierarchy of security classification levels for both confidentiality and integrity, similar to BLP. Each level represents a different degree of sensitivity or trustworthiness.

2. **Subject and Object Labels**: Assign labels to both subjects (users, processes) and objects (resources, data) indicating their respective security levels for both confidentiality and integrity.

3. **Access Control Rules**:
   - **Confidentiality Rules (BLP)**: Enforce the "no read up" and "no write down" principles to prevent unauthorized access to information at higher security levels. Subjects can read or modify objects at their own security level or lower but are restricted from accessing higher-level information.
   - **Integrity Rules (Biba)**: Apply the "no read down" and "no write up" principles to preserve the integrity of data. Subjects can only modify or access objects at their own integrity level or higher, ensuring that high-integrity data remains untampered by lower-integrity processes.

4. **Combining Rule**: Establish mechanisms to combine both confidentiality and integrity rules seamlessly. For instance, a subject with both a confidentiality and integrity level of "Secret" should only be able to access or modify objects classified at the "Secret" level for both confidentiality and integrity.

5. **Access Decision Process**: When a subject attempts to access an object, the MLS model evaluates both the subject's clearance (confidentiality level) and integrity level against the sensitivity labels of the object to determine whether the access should be granted. The access decision process should adhere to the principles of both BLP and Biba.

6. **Audit and Monitoring**: Implement auditing and monitoring mechanisms to track access attempts, modifications, and violations of both confidentiality and integrity policies. This ensures accountability and helps identify security breaches or policy violations.

7. **Adaptability and Flexibility**: Design the MLS model to be adaptable to changing security requirements and flexible enough to accommodate various scenarios, including the need for compartmentalization, role-based access, and dynamic adjustments to security levels.

By integrating elements of both BLP and Biba models into a unified MLS framework, organizations can achieve comprehensive security that addresses both confidentiality and integrity concerns across diverse computing environments. However, implementing such a model requires careful planning, rigorous enforcement, and ongoing monitoring to maintain the desired security posture.

17. BLP can be stated as "no read up, no write down." What is the analogous statement for Biba's Model?

Ans:

The analogous statement for Biba's Model can be summarized as "no read down, no write up." This means that in Biba's Model:

- **No Read Down**: Subjects are prohibited from accessing resources at a lower integrity level than their own. This prevents subjects from obtaining or viewing data that is less trustworthy than their current level of access.

- **No Write Up**: Subjects are restricted from modifying or writing data to resources at a higher integrity level than their own. This prevents subjects from introducing potentially corrupted or unauthorized data into higher integrity-level resources.

These principles are fundamental to maintaining data integrity in Biba's Model, ensuring that information flows in a controlled manner from lower integrity levels to higher ones, while preventing unauthorized modifications or contamination of trusted data.

William Stallings: Chapter 13(pg-482) Problem 1-5

Q1. The necessity of the “no read up” rule for a multilevel secure system is fairly obvious. What is the importance of the “no write down” rule?

Ans:

The "no write down" rule is equally important in a multilevel secure system, such as in the Bell-LaPadula model, for several reasons:

Prevention of Unauthorized Disclosure: By prohibiting subjects from writing or transmitting data to lower security levels, the "no write down" rule prevents the unauthorized disclosure of sensitive or classified information to less secure environments. This helps maintain the confidentiality of data by ensuring that it cannot be leaked to lower security levels where it may be accessed by unauthorized entities.

Q2. The *-property requirement for append access fc(Si) <= fo(Oj) is looser than for write access fc(Si) = fo(Oj). Explain the reason for this.

The looser *-property requirement for append access (fc(Si) ≤ fo(Oj)) compared to write access (fc(Si) = fo(Oj)) in information security models like BIBA and Bell-LaPadula is because appending data to an object does not overwrite or modify existing data. This allows controlled data flow from lower-level subjects to higher-level objects while still maintaining data integrity by preventing higher-level subjects from contaminating lower-level objects. The looser requirement strikes a balance between enabling necessary data flow and preserving the integrity of the system.

Q3. The BLP model imposes the ss-property and the *-property on every element of b but does not explicitly state that every entry in M must satisfy the ss-property and the *-property.

a. Explain why it is not strictly necessary to impose the two properties on M.

It is not strictly necessary to impose the simple security property (ss-property) and the *-property on the access control matrix (M) in the Bell-LaPadula (BLP) model because the enforcement of these properties on every element of the set of security levels (b) and the set of subjects (S) and objects (O) is sufficient to ensure the desired security properties.

The access control matrix (M) is derived from the security levels assigned to subjects and objects, as well as the access rules defined by the ss-property and the *-property. As long as these properties are enforced on the security levels, subjects, and objects, the

resulting access control matrix will automatically comply with the security requirements.

b. In practice, would you expect a secure design or implementation to impose the two properties on M? Explain.

In practice, it is generally recommended and considered a good security practice to impose the simple security property (ss-property) and the *-property on the access control matrix (M) as well, even though it is not strictly necessary from a theoretical standpoint.

Imposing these properties on the access control matrix provides an additional layer of security and helps ensure that the matrix remains consistent with the security requirements even if changes are made to the system. It also simplifies the implementation and verification of the access control mechanisms, as the properties can be checked directly against the matrix entries.

Furthermore, imposing the ss-property and the *-property on the access control matrix can help catch potential errors or inconsistencies that might have been introduced during the assignment of security levels or the derivation of the matrix from the security rules.

By enforcing these properties at both the level of security levels, subjects, and objects, as well as the access control matrix, the overall security of the system is strengthened, and the risk of inadvertent violations or vulnerabilities is reduced.

The rest of the two questions are based on a figure(very complex) in the book, That's why intentionally left. (bhison complex?? 🤣)

Bhujiye Dio amay

University Questions:

Exercise 1

The diagram below shows the access control matrix for several components of an IT system:

	File-1	File-2	File-3	File-4
User-1	r	orw	orw	–
User-2	–	–	–	–
User-3	–	r	r	orwx
User-4	orw	r	r	rx

a) Write the access control lists equivalent to this access control matrix.
b) Write the "per-subject" lists of access rights equivalent to the access control matrix.

Answer

a)
for File-1: ((User-1, r), (User-4, orw))
for File-2: ((User-1, orw), (User-3, r), (User-4, r))
for File-3: ((User-1, orw), (User-3, r), (User-4, r))
for File-4: ((User-3, orwx), (User-4, rx))

b)
for User-1: ((File-1, r), (File-2, orw), (File-3, orw))
for User-2: ()
for User-3: ((File-2, r), (File-3, r), (File-4, orwx))
for User-4: ((File-1, orw), (File-2, r), (File-3, r), (File-4, rx))

Exercise 2

Alice can read and write to file *x*, can read file *y*, and can execute file *z*. Bob can read file *x*, can read and write to file *y*, and cannot access file *z*.

a) Write a set of access control lists for this scenario. Which list is associated with which file?

b) Write the access control matrix for the system described above.

c) Write the list of subjects and the list of objects for this system.

Answer

a) file x: ((Alice, rw), (Bob, r))
file y: ((Alice, r), (Bob, rw))
file z: ((Alice, x))

b)

	file x	file y	file z
Alice	rw	r	x
Bob	r	rw	–

c) Subjects: Alice, Bob
Objects: file x, file y, file z

Exercise 3

Does the standard Unix operating system use mandatory or discretionary access control? Explain your answer.

Answer

The standard Unix operating system uses discretionary access controls, as individual users can decide which of the other system users can access their objects (e.g., files and directories) and how. The Unix command 'chmod' can be used by users who want to change the access rights for their files and directories.

Exercise 4

Consider a computer system whose access control mechanisms implement the Biba Integrity Model. Explain what you tell about the integrity levels of two processes running on this system, and which need to send, receive and process messages from each other? Justify your answer.

Answer

Two processes $p1$ and $p2$ that send (i.e., write), receive and process (i.e., read) messages from each other represent both subjects and objects within the computer system to which they belong. The Biba model needs to follow the following properties:

- The simple integrity property states that a subject at a given level of integrity must not read an object at a lower integrity level (no read down).
- The *-integrity property states that a subject at a given level of integrity must not write to any object at a higher level of integrity (no write up).

Therefore, the integrity class values for the two processes must satisfy $I(p1) = I(p2)$.

Exercise 5

The integrity levels for several subjects and objects of an information system are shown in the following table:

Integrity level	Subject	Object
4 (highest)	Alice	Personnel files
3	Bob	System log files
2	Cathy	Wiki files
1 (lowest)	Dan	Blog files

Classify EACH of the following operations as permitted or prohibited by the Biba model of security:

a) Cathy modifies the Blog files;
b) Bob updates the System log files based on information from the Wiki files;
c) Dan updates his Blog files;
d) Alice deletes a Wiki file.

Answer

The Biba model has two properties:

- Simple Integrity Property: a subject at a given level of integrity must not read an object at a lower integrity level (no read-down)
- Integrity *-Property: a subject at a given level of integrity must not write to any object at a higher level of integrity (no write-up)

a) Cathy modifies the Blog files – this means Cathy needs to write into the Blog files. This is **permitted by the *Integrity *-Property* because I(Cathy) = 2 ≥ I(Blog files) = 1;**
b) Bob updates the System log files based on information from the Wiki files – Bob needs to read the Wiki files first and then write into the System log files. The first operation is **prohibited by the *Simple Integrity Property* because I(Bob) = 3 ≥ I(System log file) = 2**;
c) Dan updates his Blog files – this is a write operation. It is **permitted by the *Integrity *-Property* because I(Dan) = I(Blog files);**
d) Alice deletes a Wiki file – if deletion is considered as a write operation, then it is **permitted by the *Integrity *-Property* because I(Alice) ≥ I(Wiki files).**

1. Can a subject in Bell-LaPadula model write to other files with different compartments but same classification?

In the Bell-LaPadula (BLP) model, a subject can write to an object if and only if the security level of the subject is greater than or equal to the security level of the object. This is known as the **"no write down"** or **"simple security property"**.

However, the BLP model also includes the concept of **compartments** (also known as categories), which are used to provide finer-grained access control. A subject can only write to an object if the subject has access to all the compartments that the object belongs to. This is known as the **"*-property"** or **"star property"**.

So, to answer your question, a subject in the BLP model **cannot write** to other files with different compartments but the same classification, unless the subject has access to all the compartments that those files belong to. This is to prevent information from flowing into compartments where it should not be. This principle is often referred to as **"no write up, no read down"**.

2. Can the Biba model always be combined with Bell-LaPadula model? What are the issues that arise?Would there be conflicts?How would information flow be affected?

The Bell-LaPadula (BLP) and Biba models are two different security models that focus on different aspects of security. The BLP model is designed to prevent information flow from a lower security level to a higher one (no read up, no write down), while the Biba model is designed to prevent information flow from a higher security level to a lower one (no write up, no read down).

In theory, these two models could be combined to create a system that prevents unauthorized information flow in both directions. However, in practice, combining these two models can lead to a situation known as a **"conflict of interest"**.

Here are some potential issues that could arise:

1. **Restrictive Access**: Combining both models could lead to a system where users have very limited access to information. For example, a user might not be able to read information from a lower level (due to BLP) and also not be able to write information to a higher level (due to Biba). This could make the system very restrictive and difficult to use.
2. **Complexity**: Implementing both models in a system would increase the complexity of the system. This could lead to increased costs and potential security vulnerabilities.
3. **Performance Impact**: The additional checks required by both models could have a negative impact on system performance.
4. **Conflicting Policies**: There could be situations where the policies of the BLP and Biba models conflict with each other. For example, a user might need to read information from a lower level and write it to a higher level as part of their job, but this would be disallowed by the combined model.

In terms of information flow, combining the BLP and Biba models would mean that information could only flow between objects of the same security level and the same set of compartments or categories. This could potentially limit the usefulness of the system, depending on the specific requirements of the organization.

3. Why is a system composed of two Bell-LaPadula-secure systems not necessarily Bell-LaPadula-secure?

The Discretionary Security Property is an access matrix that is linked to that particular system.

When you combine two systems, you must rigorously review the access matrix that will be assigned to the new composite system.

In cases where the classifications and compartments do not align perfectly, careful consideration is necessary to maintain a level of security that is equivalent or better.

In addition, there is the possibility that the scope of information available in each system is different. The aggregation of information possible after the systems are combined may exceed the intended security level, even if the original classifications and compartments aligned perfectly. The original access matrix was based on assumptions about what the information that the system would hold, and those assumptions are no longer true.

The original Bell-LaPadula model does not address either of these concerns, but any modern efforts should be capable of addressing the issues with some planning.