Post-test Knowledge Questionnaire

The following questions aim to assess your knowledge of software security and security patterns **after** your participation in the teaching and learning activities using a traditional lecture **and** the *RiskGuard* game. Please follow the recommendations below:

- Answer **exclusively based on your acquired knowledge**, without consulting materials, colleagues, or any other external sources.

- Try to answer as many questions correctly as possible. However, if you do not know the answer, select the option **"I do not know"** instead of choosing a random alternative.

Notes: Before submitting the form, please review your answers to ensure that no fields were left blank unintentionally.

1. **What is your name? *(Required)***

   ____

2. **What is your email address? *(Required)***

   ____

3. **How does an Injection attack work? *(Required)***
   ( ) An injection attack can be prevented by creating obfuscated source code, which makes scripts difficult to read and prevents any attempt to inject commands into the system.
   ( ) In an injection attack, an attacker inserts malicious commands into a vulnerable system. A strong and complex password can prevent this, as it adds an extra layer of security and makes injections harder to execute.
   ( ) An injection attack occurs when a system allows malicious data to be injected into unvalidated inputs, such as forms. Systems that properly validate and sanitize inputs can prevent these attacks, protecting data and system integrity.
   ( ) Injection attacks can only be prevented by antivirus tools that detect malicious code and automatically block any attempt to inject data into the system.
   ( ) I do not know.

4. **What are the consequences of a cryptographic failure in a web application? *(Required)***
   ( ) A security flaw that can be fixed without compromising data.
   ( ) Increased user trust due to effective cryptography.
   ( ) Improved system performance due to the use of faster algorithms.
   ( ) Leakage of sensitive data, such as passwords and personal information.
   ( ) I do not know.

5. **Correctly match the vulnerabilities with their corresponding security patterns by choosing the correct alternative: *(Required)***

Vulnerabilities	*Security Patterns*
A. SQL Injection	1. Input Validation
B. Cross-Site Scripting (XSS)	2. Output Encoding
C. Broken Authentication	3. Multi-Factor Authentication
D. Insecure Direct Object Reference	4. Access Control Enforcement

( ) 1 → B, 2 → A, 3 → D, 4 → C
( ) 1 → A, 2 → B, 3 → C, 4 → D
( ) 1 → D, 2 → C, 3 → A, 4 → B
( ) 1 → C, 2 → D, 3 → B, 4 → A
( ) I do not know.

6. **How can an Access Control Violation/Broken Access Control attack be prevented? *(Required)***
( ) Use only URLs to determine who can access what in the system.
( ) Allow anyone to access the system without restrictions.
( ) Implement Least Privilege policies and access control mechanisms.
( ) Allow users to modify their own access permissions.
( ) I do not know.

7. **Regarding SQL injection and database security: A web system stores user credentials in a database. Which of the following approaches may still be vulnerable to SQL injection, even if it appears secure? *(Required)***
( ) Use of stored procedures that do not concatenate user inputs directly into SQL commands.
( ) Validation of user input only on the frontend before sending data to the server.
( ) Use of ORM (Object-Relational Mapping) with native support for input escaping.
( ) Use of parameterized queries (prepared statements) with input type checking.
( ) I do not know.

8. **Why are logging and activity monitoring essential for system security? *(Required)***
( ) They are useful only for fiscal auditing and do not affect security.
( ) They replace the need for multi-factor authentication (MFA).
( ) They allow the identification of suspicious activities and rapid incident response.
( ) They completely eliminate the need to apply security patches.
( ) I do not know.

9. **Regarding cryptographic failures and secure credential storage: Which of the following approaches represents the best security practice for password storage? *(Required)***
( ) Create a custom password hashing algorithm, since known solutions may be

compromised.

( ) Store passwords in plaintext, protected only by a firewall and restricted database access.

( ) Use a strong hashing algorithm, such as bcrypt, Argon2, or PBKDF2, with a unique salt for each password.

( ) Use the same encryption key across multiple systems.

( ) Store passwords encrypted with AES-256 using a fixed key embedded in the source code.

( ) I do not know.

10. **Regarding broken access control: Which of the following configurations can lead to a broken access control vulnerability in a web system? *(Required)***

( ) Implementing access control lists (ACLs) to define role-based permissions.

( ) Implementing access control only on the frontend without backend enforcement.

( ) Applying the principle of least privilege to all administrative functions.

( ) Using JSON Web Tokens (JWT) with a short expiration time.

( ) I do not know.

11. **After an attack on an e-commerce platform in which customer payment data were altered by an attacker, what would be the best practice to prevent data integrity failures? *(Required)***

( ) Allow customers to modify their own transactions directly in the system without additional controls.

( ) Implement strong encryption for sensitive data (e.g., credit card numbers) and use multi-factor authentication for administrative access.

( ) Ignore data integrity validation when receiving data from third parties, such as payment processors.

( ) Do not encrypt payment data so employees can easily access them.

( ) I do not know.

12. **In a company, a version control system was compromised due to a lack of code authenticity verification. What could be done to improve software integrity and protect against unauthorized modifications? *(Required)***

( ) Store source code on public servers to ensure easy and fast access.

( ) Implement digital signatures on source code and use a version control system that records all changes with strict auditing.

( ) Avoid applying patches or updates to prevent integrity risks.

( ) Allow any employee to modify the source code without control or approval.

( ) I do not know.

13. **On an e-commerce website, a user changes the account ID in the URL from “12345” to “12346” and accesses another user’s data. Which security vulnerability is occurring? *(Required)***

( ) CSRF

( ) Broken Access Control

( ) SQL Injection

( ) XSS

( ) I do not know.

14. **An attacker exploits a flaw in a software system by altering source code or stored data to modify system behavior, compromising its accuracy and reliability. Which vulnerability is being exploited in this scenario?** ***(Required)***

    ( ) Code or data tampering (exploiting software integrity)
    ( ) Cross-Site Scripting (XSS)
    ( ) Phishing
    ( ) SQL Injection
    ( ) I do not know.