NAMIBIA UNIVERSITY
OF SCIENCE AND TECHNOLOGY
Faculty of Computing and Informatics

Department of Cyber Security

QUALIFICATION : Bachelor of Computer Science in Cyber Security	
QUALIFICATION CODE: 07BCCS	**LEVEL:** 6
COURSE: Web Application Security	**COURSE CODE:** WAS621S
DATE: November 2025	**PAPER:** Theory
DURATION: 180 minutes	**MARKS:** 100

FIRST OPPORTUNITY EXAMINATION QUESTION PAPER	
EXAMINER(S)	**Ms. V. Shakela** **Mr. Adriaan Grobler** **Mr. Petrus Katambo** **Mr. Andreas Amukwa**
MODERATOR:	**Mr. E. Nepolo**

THIS QUESTION PAPER CONSISTS OF 6 PAGES
(Excluding this front page)

INSTRUCTIONS

1. Answer ALL the questions.
2. Write clearly and neatly.
3. Number the answers clearly.
4. When answering questions, you should be guided by the allocation of marks. Do not give too few or too many facts in your answers.

PERMISSIBLE MATERIALS

1. Non-programmable calculator

Section A **[20 Marks]**

Multiple Choice (15 marks)

1. You have discovered that ports 80 and 8080 are open on the web server during the reconnaissance stage. What does this mean about the web application security posture?
   A. The application is vulnerable to SQL injection attacks by default.
   B. The server likely hosts HTTP services, which may expose web traffic to interception attacks.
   C. The web server is secured if these ports are open and traffic is encrypted.
   D. Ports 80 and 8080 are closed, hence web requests to this ports will be terminated.

2. Reconnaissance findings reveal numerous URLs with exposed sensitive parameters (e.g., session tokens in URLs). What is the primary risk in this situation?
   A. Exposure to URL parameter manipulation.
   B. URLs do not carry sensitive data, hence no security risk.
   C. Parameters in URLs improve application security.
   D. Session tokens in URLs are automatically encrypted.

3. Which of the following scenarios would NOT be considered a secure context in web browsing?
   A. A page loaded with a valid SSL certificate from a verified CA.
   B. A page loaded on https: //localhost.
   C. A page loaded over HTTP with a GET request method.
   D. A page loaded from a browser extension.

4. The purpose of site Isolation, as part of the browser security model is to protect web users through:
   A. Allowing websites to share cookies across domains as long as they have implemented cookie security attributes.
   B. Disabling JavaScript on all web pages
   C. Encrypting all website traffic
   D. Running different websites in separate processes so they cannot access each other's data

5. Which of the following mechanisms enforces trusted sources for content loading?
   A. Content Security Policy (CSP)
   B. Same-Origin Policy (SOP)
   C. Secure Socket Layer (SSL)
   D. Cross-Origin Resource Sharing (CORS)

6. Hackers often gather a multitude of seemingly small, innocuous pieces of configuration about a website that, when combined, can help exploit the site. Which of the following error messages is typically considered NOT safe to display to the user?
   A. A message that states that the system is down for maintenance and tells what time it is expected to be back up. E.g.: Our site is down. We're sorry for the inconvenience. We are doing maintenance on our servers. The site should be up by 12h00.
   B. An error message that says there was an internal error message and displays the call stack to assist in debugging and reporting of the error. E.g.: There was an internal error, please copy and paste this page to the sysadmin.
   C. A message that says that there was an error logging in mentioning the username. E.g.: User "JoeUser" could not be logged in with the information you provided.
   D. An error message that says there was an internal error but does not provide any details to assist in debugging or reporting of the error. E.g.: There was an internal error. Please report this to the sysadmin.

7. Which of the following techniques is an effective way to mitigate against SQL Injection attacks on a web application?
   A. Using prepared statements.
   B. Using Username and password authentication method.
   C. Reducing the amount of data in HTTP responses.
   D. Implementing strict password policies.

8. How does the server validate authentication in token-based authentication?
   A. Looks up the session ID in a database
   B. Decrypts and verifies the token's signature
   C. Matches the token with a user record in the server database
   D. Relies on cookies sent by the client

9. The attacker has intercepted the user's session and was able to retrieve the session token. What is likely to happen while the attacker is in possession of a valid token?
   A. The attacker can perform session hijacking
   B. The server will refuse the valid token
   C. The server immediately invalidates the token
   D. Tokens self-destruct instantly after interception

10. Which method allows immediate server-side user logout by invalidating session data?
   A. Token-based authentication
   B. Session-based authentication
   C. Both token and session-based authentication

D. Multi-factor authentication (MFA)

11. What is the primary purpose of the Strict-Transport-Security (HSTS) header in HTTP communication?
    A. To allow cross-origin resource sharing (CORS)
    B. To specify allowed content sources for scripts and images
    C. To enable Cross Site Scripting (XSS) filtering in browsers
    D. To enforce that browsers only connect to the server over HTTPS

12. Which of the following is NOT acceptable as a guideline to writing secure codes in web applications?
    A. Storing Passwords as ciphertext
    B. Using hardcoded credentials in your code
    C. Writing a code that handles errors to prevent a program from crashing.
    D. Filtering and validating user input data

13. Which of the following HSTS headers will disable the HSTS rule on the browser?
    A. Strict-Transport-Security: max-age=none
    B. Strict-Transport-Security: max-age=disable
    C. Strict-Transport-Security: max-age=0
    D. Strict-Transport-Security: max-age= “”; includeSubDomains; preload

14. In which way does machine learning make modern web application firewalls (WAF) more effective?
    A. It enables them to respond effectively to the continuously evolving threat landscape
    B. It allows WAFs to return search results quicker than using traditional filtering methods.
    C. Machine learning enables the WAFs to allow insecure APIs
    D. It allows them to choose the most suitable web application for a specified task

15. How does the Content-Security-Policy (CSP) header enhance web application security?
    A. By blocking all scripts and images loading from different origins
    B. By restricting which domains are allowed to load content such as scripts
    C. By forcing all content to be loaded over HTTP connections
    D. By disabling browser caching to ensure client-side security

True/False (5 marks)

1. A Cross-Site Request Forgery (CSRF) attack occurs when an attacker injects malicious JavaScript into web pages that are then executed in other web users' browsers.

2. The Secure Development Lifecycle integrates security activities into every phase of the software development process.
3. SQL Injection vulnerabilities allow attackers to execute arbitrary SQL commands by inserting malicious input into an application's database queries.
4. Cross-Origin Resource Sharing (CORS) is an HTTP-header-based mechanism that instructs a server to deny loading resources from any origins other than its own.
5. Insecure Direct Object References (IDOR) occur when applications expose internal objects without proper access control, allowing attackers to access unauthorized data.

Section B **[80 Marks]**

Question 1 [30 Marks]

1.1. Discuss the differences between passive and active reconnaissance techniques in web application security assessment and provide one suitable reconnaissance tool that can be used for each technique. **(6)**

1.2. Illustrate with an example how a browser isolates scripts and data from different origins to prevent unauthorised data access within the same browsing session. **(6)**

1.3. Cross Origin Resource Sharing (CORS) headers allows controlled resource sharing between websites. Discuss the key security considerations that web developers must enforce when configuring CORS policies. **(6)**

1.4. Explain how the JavaScript document.domain property can be used to relax the Same-Origin Policy (SOP) between subdomains. Provide a suitable example to motivate your explanation. (5)

1.5. Describe a server-side defenses that mitigates the effects of brute force (testing multiple passwords from a dictionary against a single account), credential stuffing (testing username/password pairs obtained from a breach), as well as password spraying (testing a single weak password against a large number of different accounts). **(4)**

1.6. Identify three critical factors that enable a Cross-Site Request Forgery attack to succeed. (3)

Question 2 [20 Marks]

Examine the image below and answer the following questions

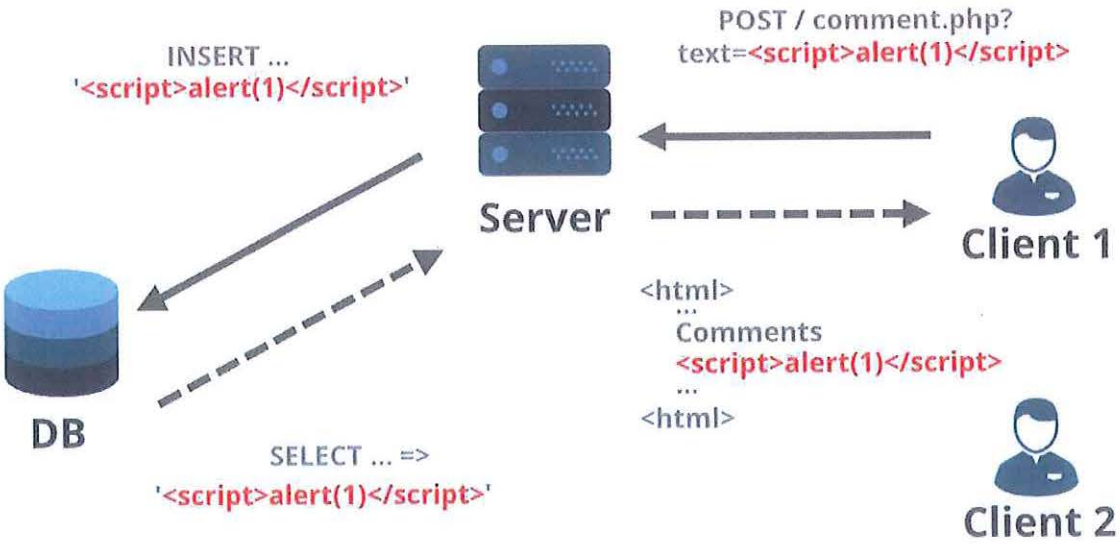

Figure 1

2.1. What type of web attack is demonstrated in Figure 1? Justify your answer. **(4)**

2.2. Based on figure 1, outline the step-by-step attack flow as presented in the diagram. **(10)**

2.2. Explain three (3) effective security measures to protect against the web attack demonstrated in figure 1. **(6)**

Question 3 **[15 Marks]**

Study the mini case below carefully and answer the following questions.

An e-commerce website allows users to log in to their accounts to manage orders, payment information, and personal details. The web application uses session cookies to maintain user login states. During a security test, it is discovered that session tokens are predictable, do not expire promptly after logout or inactivity and are transmitted over HTTP connections in some parts of the application. An attacker exploited these weaknesses by using automated tools to predict valid session tokens and hijack active user sessions without needing to re-authenticate.

As a result, multiple user accounts were exploited, leading to unauthorised purchases, leakage of sensitive financial information, and substantial reputational damage to the company.

3.1. From the case study, identify and discuss three (3) technical limitations in the session authentication process that contributed directly to the exploitation of user accounts. **(6)**

3.2. Mention and explain three cookie security flags that can be implemented to protect session cookies against unauthorised access. **(6)**

3.3. Explain why it is important to invalidate a user session on logout in web applications. **(3)**

Question 4 **[15 Marks]**

4.1. Explain the difference between vertical and horizontal privilege escalation. Provide examples of how each might be exploited in a web application. **(6)**

4.2. Analyse the potential security risks of relying solely on client-side access control enforcement. How can attackers exploit this weakness? **(5)**

4.3. Differentiate between Static Application Security Testing (SAST) Dynamic Application Security Testing (DAST) **(4)**