BRAINWARE UNIVERSITY

Term End Examination 2024-2025
Programme – B.Sc.(ANCS)-Hons-2023
Course Name – Web Application Penetration Testing
Course Code - BNC40112
(Semester IV)

Time : 2:30 Hours

Full Marks : 60

[The figure in the margin indicates full marks. Candidates are required to give their answers in their own words as far as practicable.]

Group-A
(Multiple Choice Type Question) 1 x 15=15

1. *Choose the correct alternative from the following :*

(i) Recall the protocol primarily used for real-time web communication.
a) REST b) WebRTC
c) WebSockets d) GraphQL

(ii) Recognize the scripting language that adds interactivity to a webpage.
a) HTML b) CSS
c) JavaScript d) Assembly

(iii) Identify the tool used for securing communication in HTTPS.
a) SQL b) TLS
c) FTP d) SSH

(iv) What does Cross-Site Scripting (XSS) allow attackers to do?
a) Inject malicious scripts into web pages b) Execute server-side scripts remotely
c) Access encrypted HTTPS traffic d) Disable firewall protection

(v) Which command helps validate XSS vulnerabilities in Burp Suite?
a) SQL Injection Scan b) HTTP Request Interception
c) DOM Scanning d) URL Encoding Tool

(vi) Evaluate the impact of DOM-Based XSS on client-side security.
a) Compromises only network security b) Compromises browser behavior
c) Steals server credentials d) Hijacks admin credentials

(vii) How does HTML entity encoding mitigate XSS?
a) Converts special characters to safe representations b) Validates cookies
c) Prevents phishing attacks d) Encodes URL parameters

(viii) Identify how an attacker could exploit UNION-based SQL Injection.
a) By appending UNION SELECT username, password FROM users b) By using ORDER BY to arrange data

c) By encrypting SQL queries
d) By limiting database user privileges

(ix) Select the SQL function used in time-based Blind SQL Injection.
a) SLEEP()
b) CONCAT()
c) COUNT()
d) DISTINCT()

(x) Define different payloads to test for SQL Injection vulnerabilities.
a) ' OR '1'='1
b) SELECT * FROM users WHERE username = 'safeuser'
c) DELETE FROM users WHERE id=1
d) CREATE INDEX user_index ON users(username)

(xi) Choose the best security measure for preventing session hijacking.
a) Implementing token-based authentication
b) Storing session IDs in local storage
c) Using weakly encrypted session identifiers
d) Sharing session cookies across domains

(xii) Identify the importance of regenerating session IDs after authentication.
a) Prevents attackers from reusing stolen session tokens
b) Allows sessions to persist indefinitely
c) Increases application performance
d) Ensures session tokens are shared across multiple users

(xiii) Define the role of session encryption in preventing hijacking attempts.
a) Ensures session data remains unreadable if intercepted
b) Prevents session expiration
c) Disables cookie storage in the browser
d) Increases server response time

(xiv) Predict the best approach to prevent file upload abuse.
a) Preventing users from accessing file upload forms
b) Storing uploaded files in executable directories
c) Implementing strict allowlists and server-side validation
d) Allowing all file types to be uploaded

(xv) Justify the need for sandboxing in handling uploaded files.
a) Reduces the number of files stored on the server
b) Ensures potentially malicious files are isolated before execution
c) Increases database storage efficiency
d) Prevents file downloads from external sources

Group-B
(Short Answer Type Questions) 3 x 5=15

2. Examine the role of SameSite cookies in session security. (3)
3. Explain the request-response cycle in web applications. (3)
4. Describe the role of client-side JavaScript in DOM-Based XSS. (3)
5. Explain the concept of SQL Injection. (3)
6. Develop a file handling strategy to prevent execution vulnerabilities. (3)

OR

Design a comprehensive security framework to mitigate file-based attacks. (3)

Group-C
(Long Answer Type Questions) 5 x 6=30

7. Explain the difference between Reflected XSS and Stored XSS. (5)
8. How does a Web Application Firewall (WAF) contribute to web security? (5)
9. Compare session-based and token-based authentication models. (5)
10. Evaluate the differences between Local File Inclusion (LFI) and Remote File Inclusion (RFI). (5)
11. Compare the security risks of Server-Side Request Forgery (SSRF) and Cross-Site Request Forgery (CSRF). (5)
12. Assess the effectiveness of Web Application Firewalls (WAF) against SQL Injection. (5)

OR

Evaluate the risks associated with Time-Based Blind SQL Injection. (5)

****