REG.NO.:

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering
CONTINUOUS ASSESSMENT TEST - II
FALL SEMESTER 2025-2026

SLOT: B2+TB2

Programme Name & Branch	: B.Tech CSE with Specialization in Information Security
Course Code and Course Name	: Web Application Security - BCSE320L
Faculty Name(s)	: Dr. Arulkumar V, Dr. Samriddhi Sarkar
Class Number(s)	: 1454, 1459
Date of Examination	: 06-Oct-2025
Exam Duration	: 90 minutes

Maximum Marks: 50

General instruction(s):

- Answer All Questions
- M - Max mark; CO – Course Outcome; BL – Blooms Taxonomy Level (1 – Remember, 2 – Understand, 3 – Apply, 4 – Analyse, 5 – Evaluate, 6 – Create)
- Course Outcomes
  2. Familiarize the Application Programming Interface analysis and vulnerability management of securing a web-based system
  3. Learn the web application hacking techniques and prevention solutions.

Q. No	Question	M	CO	BL
1.	A financial technology (VINtech) startup is rapidly adopting new frameworks and third-party libraries to speed up development of its web platform. As the tech stack evolves, the team becomes concerned about the risk of security vulnerabilities introduced via these external dependencies. Explain how using the Common Vulnerabilities and Exposures (CVE) database can help this team identify and mitigate risks associated with the adoption and reinvention of new technologies. Illustrate with examples how regular vulnerability assessment using CVE references can prevent incidents such as injection attacks that often target modern frameworks.	10	2	2
2.	A social media web application allows users to search for other users and post comments. During a security review, concerns arise about potential XSS vulnerabilities. Analyze the Reflected XSS, DOM-Based XSS, and Mutation-Based XSS could each be exploited in this scenario. Highlight the differences in their attack vectors, explain the role of server-side versus client-side code in these attacks, and suggest how detection and mitigation strategies would vary for each type?	10	2	4
3.	A developer is building a web-based payment portal where users can transfer money and update account information. The application relies heavily on POST requests and query parameters for data transmission between client and server. List and describe common security threats related to query parameter tampering and CSRF (Cross-Site Request Forgery) against POST endpoints in relevance to the following aspects: • Define query parameter tampering and provide real-world examples of the types of information attackers may target. • Define CSRF attacks and explain how POST requests can be vulnerable in web applications. • List effective prevention strategies for each threat, mentioning practices such as input validation, session tokens, CSRF tokens, encryption, and the use of secure communication channels.	10	3	2
4.	A travel booking website allows users to register, search, and book itineraries. The site processes user-provided data in its queries and system commands but has not been securely coded against injection attacks. During a security	10	3	4

	assessment, multiple types of vulnerabilities are discovered, including SQL injection, code injection, and command injection. Analyze the injection vulnerabilities can be exploited on the web platform by attackers and for each type of SQL injection, code injection, and command injection in the follow aspects: • Describe the general exploitation technique • Give a relevant example specific to a travel booking website (such as manipulating booking records, executing unauthorized code, or issuing system-level commands), • Analyze and discuss the immediate and long-term security consequences • Outline suitable preventive controls, the development team should apply in each case.			
5.	A cloud-based web application experiences multiple denial-of-service (DoS) attacks, including one caused by maliciously crafted regular expressions (ReDoS), another exploiting logical flaws in its request handling (Logical DoS), and a widespread attack leveraging many distributed sources (Distributed DoS). Explain the key differences between these types of DoS attacks and illustrate how each attack impacts the application's availability and performance. Discuss mitigation strategies an engineering team could implement to defend against each variant, including practical coding safeguards for ReDoS vulnerabilities, request validation for Logical DoS, and network-level protections for Distributed DoS.	10	3	2

****