



Web Application Security Assignment

Target Website (for study & observation only):

`http://paul.nasawebtech.com/`



Assignment Rules

- Internet usage is allowed for:
 - Understanding concepts
 - Referring documentation
 - Researching attack types & prevention techniques
 - Do **NOT** perform unauthorized attacks on the live website
 - This assignment is based on:
 - Payload analysis
 - Behavior observation
 - Theory + controlled lab testing
 - Screenshots must be taken **only from lab environment**
-



Section 1: Application Reconnaissance (Observation Only)

Task 1: Website Understanding

1. Open the given website in browser.
2. Identify:
 - Input fields (login, search, forms)
3. Write:
 - Which input looks **user-controlled**
 - Why user input is risky

! *Do not inject payloads on live site.*



Section 2: SQL Injection (Analysis + Lab Simulation)

Task 2: SQL Injection Theory

1. What is **SQL Injection**?
2. Why input validation failure causes SQL Injection?

3. Explain the payload:

```
' OR '1'='1
```

4. What happens at database level when this payload is executed?

Task 3: SQL Injection – Simulated Practical

1. Identify login form in given website.

2. Try:

- Normal credentials
- SQLi payload (`' OR '1'='1`)

3. Observe:

- Login bypass behavior

4. Take screenshots of:

- Login page
- Successful/failed attempt

5. Write:

- Why authentication was bypassed
-

Section 3: Other Common Web Attacks (Theory + Mapping)

Task 4: Attack Mapping (No Execution on Live Site)

For **each attack**, answer the following:

Attack Type	Description	Example Payload	Impact
SQL Injection			
XSS			
Brute Force			

 Payloads are **for study only**, not execution on live site.

Section 4: Logs & Detection (Blue Team Angle)

Task 5: Log Identification

1. Which logs would record SQL Injection attempts?

- Apache access log

- Error log
-

Task 6: Brute-force Detection

1. How does a brute-force attack appear in logs?
 2. Which tools help prevent it?
 3. Role of:
 - Fail2ban
 - Rate limiting
-

Section 5: Incident Response Scenario

Task 7: Incident Response Case Study

Scenario:

A website shows multiple SQL Injection attempts in Apache logs.

1. Identify the incident
 2. Map it to Incident Response phases:
 - Identification
 - Containment
 - Eradication
 - Recovery
 - Lessons Learned
 3. Write actions taken in each phase
-

Section 6: Attack Intelligence

Task 8: Attacker Intelligence (Theory)

Study following websites:

- haveibeenpwned.com
 - intelx.io
 - iknowwhatyoudownload.com
-

Section 7: Security Mitigations

Task 9: Prevention Techniques

1. How to prevent SQL Injection?

2. Importance of:

- Prepared Statements
- Input validation
- WAF

3. How logs help in early detection?



Submission Instructions

Upload to GitHub:

- Screenshots (lab only)
- Text files (answers & logs)