

Forum: [Hack Forums](#)

Thread: [SQL Injection: A Complete Course part two](#)

ICTKUN - September 20, 2026, 1:35 am

3.5 Defensive lesson

Every input point on that list is a place where a developer made a trust decision. The defender's job is to know every one and prove each is parameterized. If you can't produce that list for your own app, you can't tell me it's not injectable. Recon isn't an attacker-only skill — it's the first step of any audit.

Part 4 — Testing (Confirmation, Not Exploitation)

You're answering one question: does this input reach the database as code?

You are not dumping yet. You're confirming.

4.1 DVWA, Low security

Go to SQL Injection. Enter 1. You get a user back. Normal.

Now enter:

Code 1'

You get a SQL error. Confirmed. The input is being concatenated into a query and the quote broke it.

Mechanically, the app ran:

Code `SELECT first_name, last_name FROM users WHERE user_id = '1'`

The trailing ' is unbalanced. The DB parser choked. The error came back to the browser because DVWA is deliberately misconfigured. Two bugs in one: string concatenation and verbose error leakage.

Now enter:

Code 1' --

(trailing space after -- matters in some DBs)

You get user 1 back, no error. You closed the quote and commented out the rest. Confirmed.

4.2 Column count

To pull data with UNION, you need the same number of columns as the original query.

Code 1' ORDER BY 1 --

1' ORDER BY 2 --

1' ORDER BY 3 --

When you hit an error, you've exceeded the column count. DVWA's query has 2. So:

Code 1' UNION SELECT 1, 2 --

If you see 1 and 2 on the page, you have a working UNION injection with 2 columns.

4.3 Juice Shop

Code /rest/products/search?q=

Code Try q=apple — normal. Now q=apple' --. A 500 or DB error in the JSON response confirms it. Juice Shop returns SQLite errors that often leak the query.

Column count: try UNION SELECT 1,2,3,4,5,6,7,8,9 -- and adjust until it stops erroring. Standard Juice Shop search has 9 columns.

4.4 PortSwigger labs

Do these in order — cleanest teaching progression on the internet:

SQL injection vulnerability in WHERE clause allowing retrieval of hidden data

SQL injection vulnerability allowing login bypass

SQL injection UNION attack, determining the number of columns returned by the query

SQL injection UNION attack, finding a column containing text

SQL injection UNION attack, retrieving data from other tables

Each lab gives you a target, a goal (e.g., "log in as administrator"), and a solution checker. Work them manually. Don't paste payloads from the solution until you've tried.

4.5 Blind confirmation

DVWA's Blind module gives no output. Confirm with boolean logic:

Code 1' AND '1'='1 -> page loads normally (true)

1' AND '1'='2 -> page loads differently (false)

Different responses = boolean oracle. You can extract data one bit at a time.

Time-based:

? loading encrypted image...

5-second delay = injected SQL executed. That's your channel when nothing else shows.

4.6 Defensive lesson

Code Confirmation technique What made it possible The fix

1' -> error String concatenation + verbose errors Parameterize; generic error pages

1' -- -> bypass Concatenation Parameterize

ORDER BY column count Concatenation + output echo Parameterize; don't echo raw results

Boolean blind Concatenation + distinguishable responses Parameterize; uniform error handling

Time blind Concatenation + no rate limiting Parameterize; rate limit; monitor latency

Notice the pattern. Every single one is fixed by parameterization. The others are defense in depth. The bug is one bug.

Part 5 — Dumping (Extraction, In the Lab)

Now you extract. In a real engagement you'd stop at "confirmed" and report. In the lab you dump, because seeing the data leave teaches you what's at stake.

5.1 DVWA — UNION dump

You have 2 columns. Start with metadata:

Code 1' UNION SELECT user(), database() --
DB user and DB name. Now the schema:

Code 1' UNION SELECT table_name, column_name FROM information_schema.columns WHERE
table_schema = database() -- Posting part 3 into a moment !

Part 3 is here : <https://hackforums.net/showthread.php?tid=6332177>