Assignment 4

SQL Injection

For this assignment you are given a login page, CTF-style, application. This application uses a database that has 2 tables. A table with “users” and a table with “items”. The application has 2 functionalities:

- Login:
    - As “user”
    - As Administrator with the username “superadmin”
- Search and display the information about an item given its **exact** name.

Your goal is use the SQL Injection Vulnerabilities present withing the application’s login to login as “superadmin”!

Tasks

1. Bypass the login page using an SQL injection query and log in as “user”
2. When logged in use the search functionality to retrieve information from the ‘users’ table instead of ‘items’
3. Find the “superadmin’s” password and login in to the admin dashboard.

How to Run locally

1. Download the given zip, extract and `cd public/`
2. Make sure you have python v3 installed
3. Run `python3 -m pip install -r requirements.txt`
4. Run `./run.sh`
5. The application will be available under `http://127.0.0.1:8080`

Notes

1. The remote application is running on this IP address: `http://140.238.219.114:1337`, and will remain online for the duration of the exersice
2. The application has 3 pages and 3 different URL paths:
    a. User login -> (“/”)
        i. User Dashboard -> (“/dashboard”)
    b. Admin login -> (“/admin”)

    You won’t be able to access the dashboard without logging in first!
3. You are being provided the mock code for this assignment, meaning you have the full working source code which we encourage you to run locally in order to see what’s going on behind the

scenes. However the passwords stored in the database are fake. Do not use them to login in the real application! (mock credentials are Login:Password “superadmin:superadmin”, “user:user”)

4. As soon as you are able to exploit it locally, give it a go on the real server
5. For logging in the username is being set automatically, you need only enter the password for the account
6. The two tables with the column names are given below:

Table: **users**

ID (primary key)	**username**	**password**
0	superadmin	XXXXXXXXXXXX
1	user	XXXXXXXXXXXX

Table: **items**

ID (primary key)	**name**	**category**	**price**
0	item0	Music	123
….	-//-	-//-	-//-
99	item99	Music	123

Do’s and Do not’s

1. **Run the application locally and examine the queries.**
2. **The only file you need to examine is `app/app.py`, don’t bother with any other!**
3. **Examine `app.py` carefully and use the comments for guidance!**
4. **Do not edit anything in the ‘db’ folder!**
5. **Do not brute force or use any other tool that automatically initiates requests! YOU ARE REQUEST LIMITED! (500/hour and 1/sec)**

Deliverables

You need to deliver a README that explains you process to solve every task, as well as the input for the 2 fields.