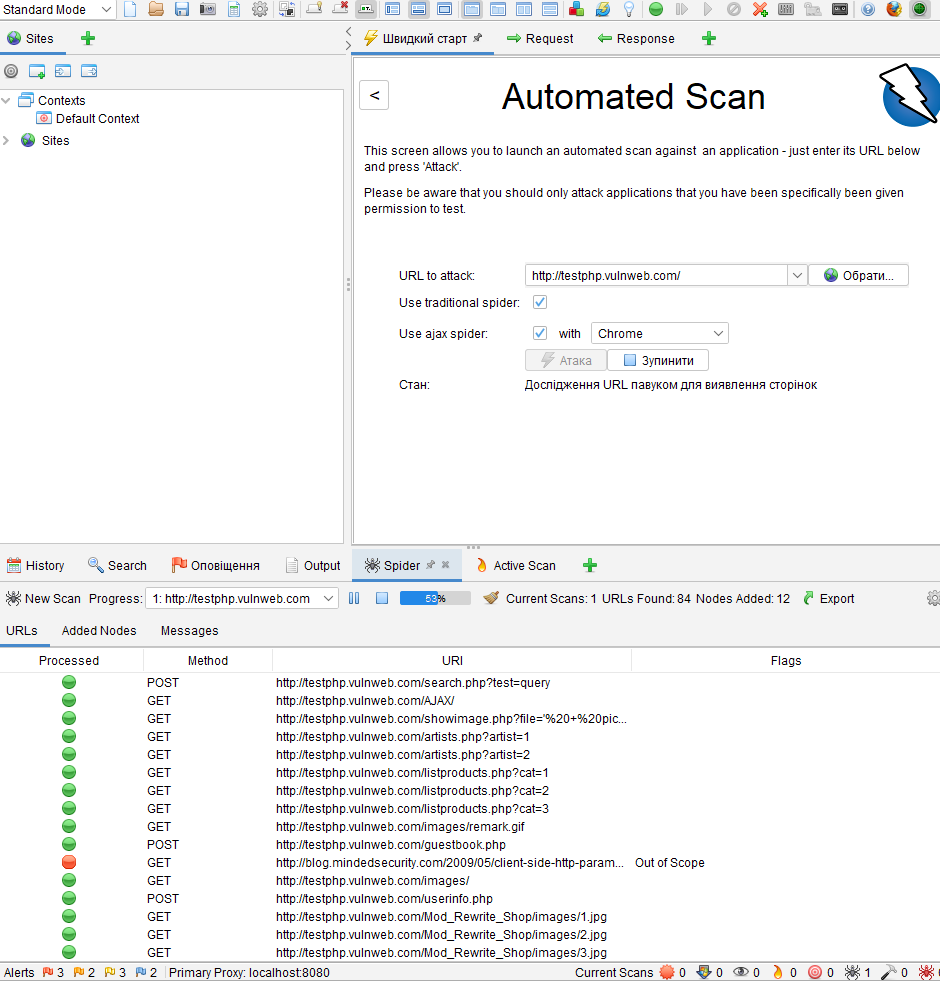
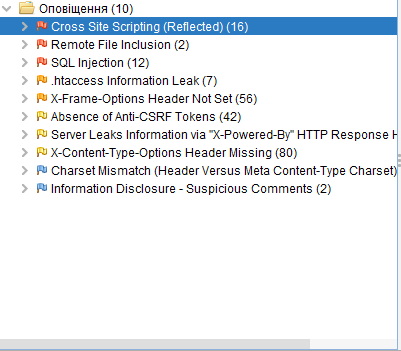
**Practice 14 Svyatoslav Fedyk (OWASP)**

**Запуск програми OWASP та атака на сайт:**

Site: <http://testphp.vulnweb.com/>

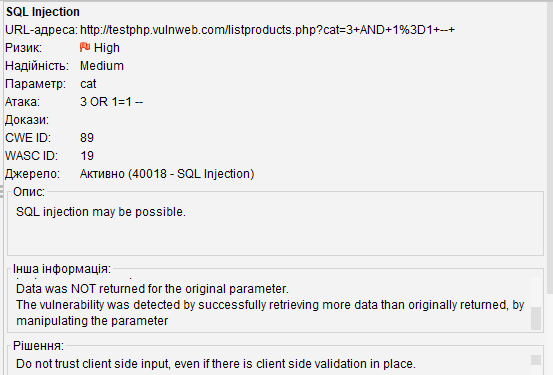


В результаті ми отримали велику кількість SQL, XSS та RFI(PHP injection) вразливостей.

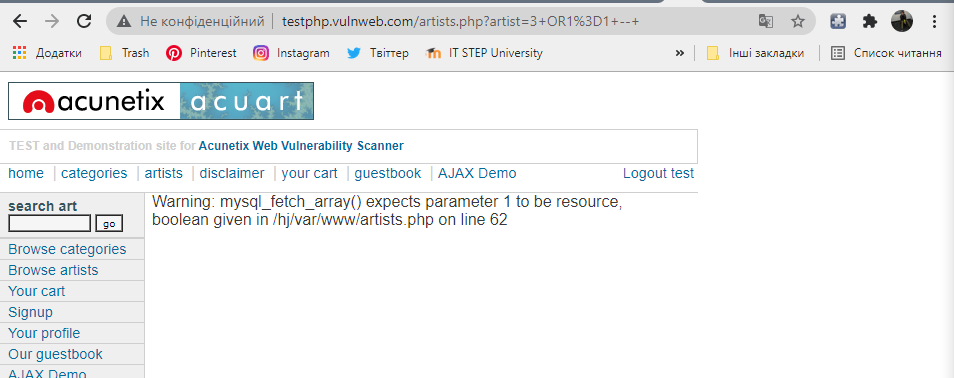
****

**Перевірка вразливостей :**

**SQL Injection:**

****

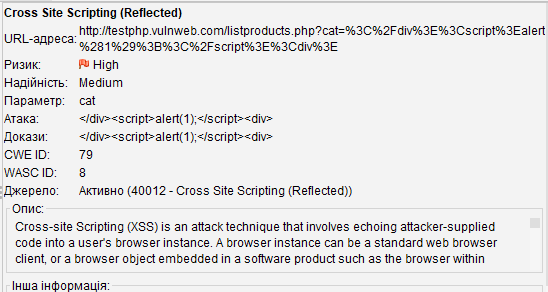
Attack: [http://testphp.vulnweb.com/artist.php?cat=3+OR+1%3D1+--](http://testphp.vulnweb.com/artist.php?cat=3+AND+1%3D1+--)+



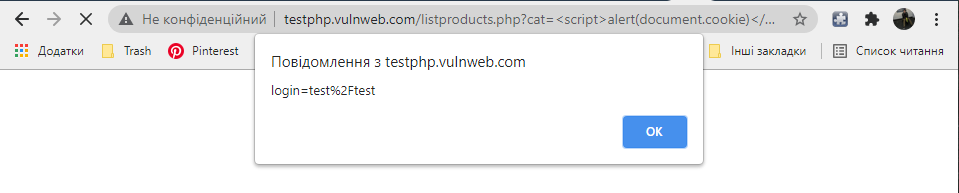
В даному випадку ми зруйнували логіку сторінки, але за допомогою SQL injection можна отримувати данні з бази даних, котрі простий юзер не повинен бачити або знищити їх з допомогою DROP TABLE.

Отже, не довіряйте введенню на стороні клієнта, навіть якщо існує перевірка на стороні клієнта. Загалом, введіть перевірка всіх даних на стороні сервера. Якщо програма використовує JDBC, використовуйте PreparedStatement або CallableStatement з параметрами.

**XSS:**

****

Attack: http://testphp.vulnweb.com/listproducts.php?cat=%3Cscript%3Ealert(document.cookie)%3C/script%3E



Як ми бачимо на скріні атака працює і це є великим ризиком, адже зловмисники можуть добавити свої скрипти в посилання, що може призвести до втрати важливих даних, якщо користувач перейде по цьому посиланню.

Рішенням може бути використання перевіреного та актуального фреймворку, котрий не дозволяє виникнути цій слабкості, або надають інструменти, які полегшують уникнення цієї слабкості.