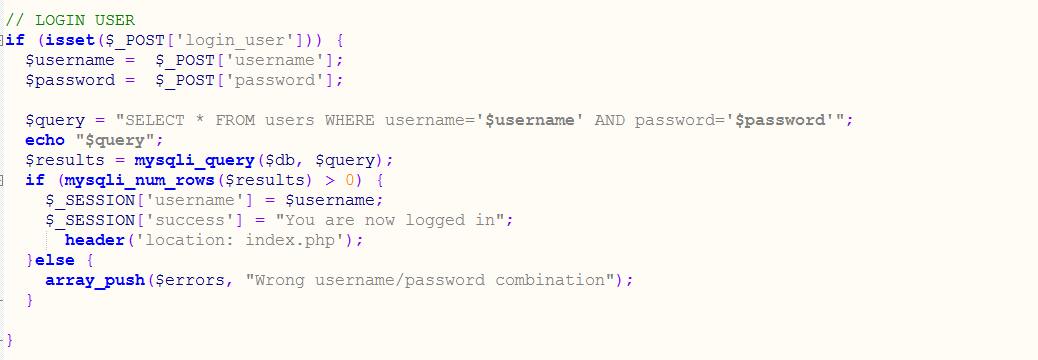
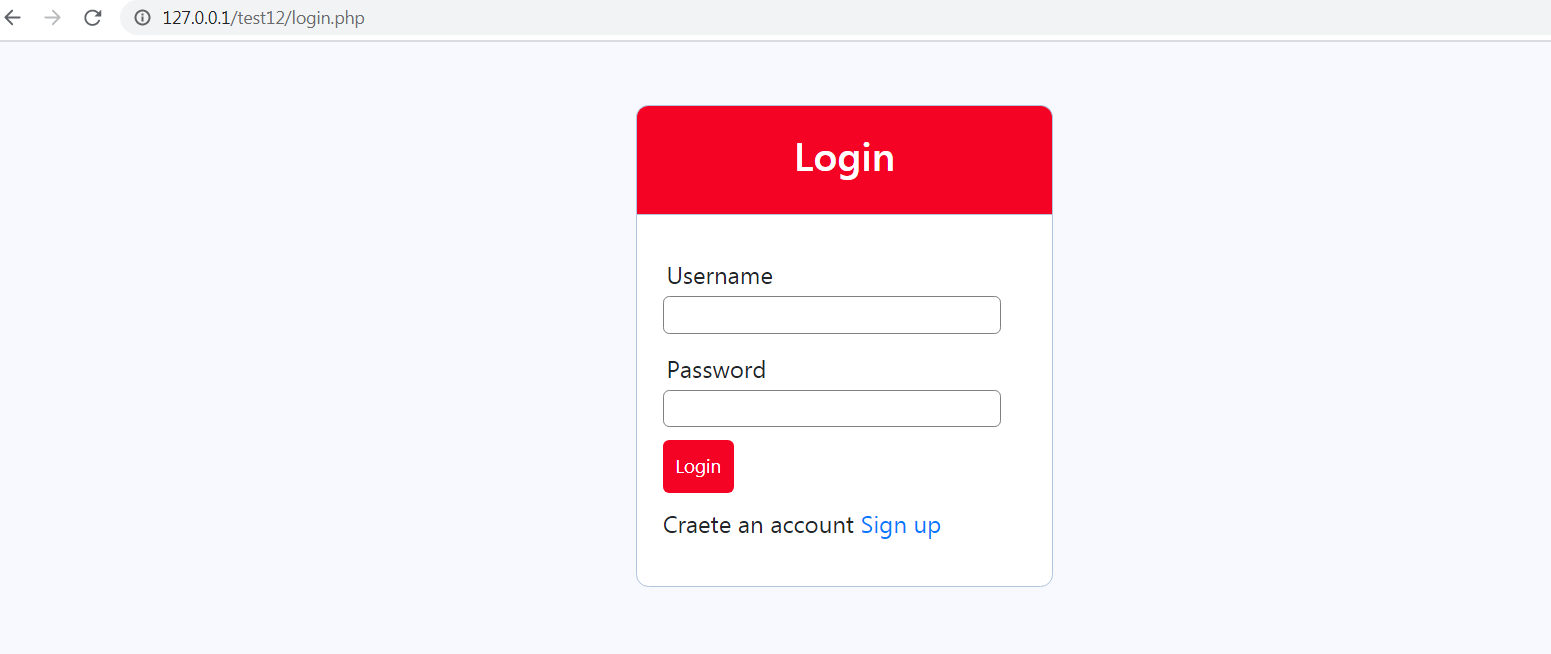
1. **MySQL**

Clone code trên github về cài đặt trên máy ảo win 7 với MySQL XAMPP. Sửa lại query SQL ta có đoạn code như sau:

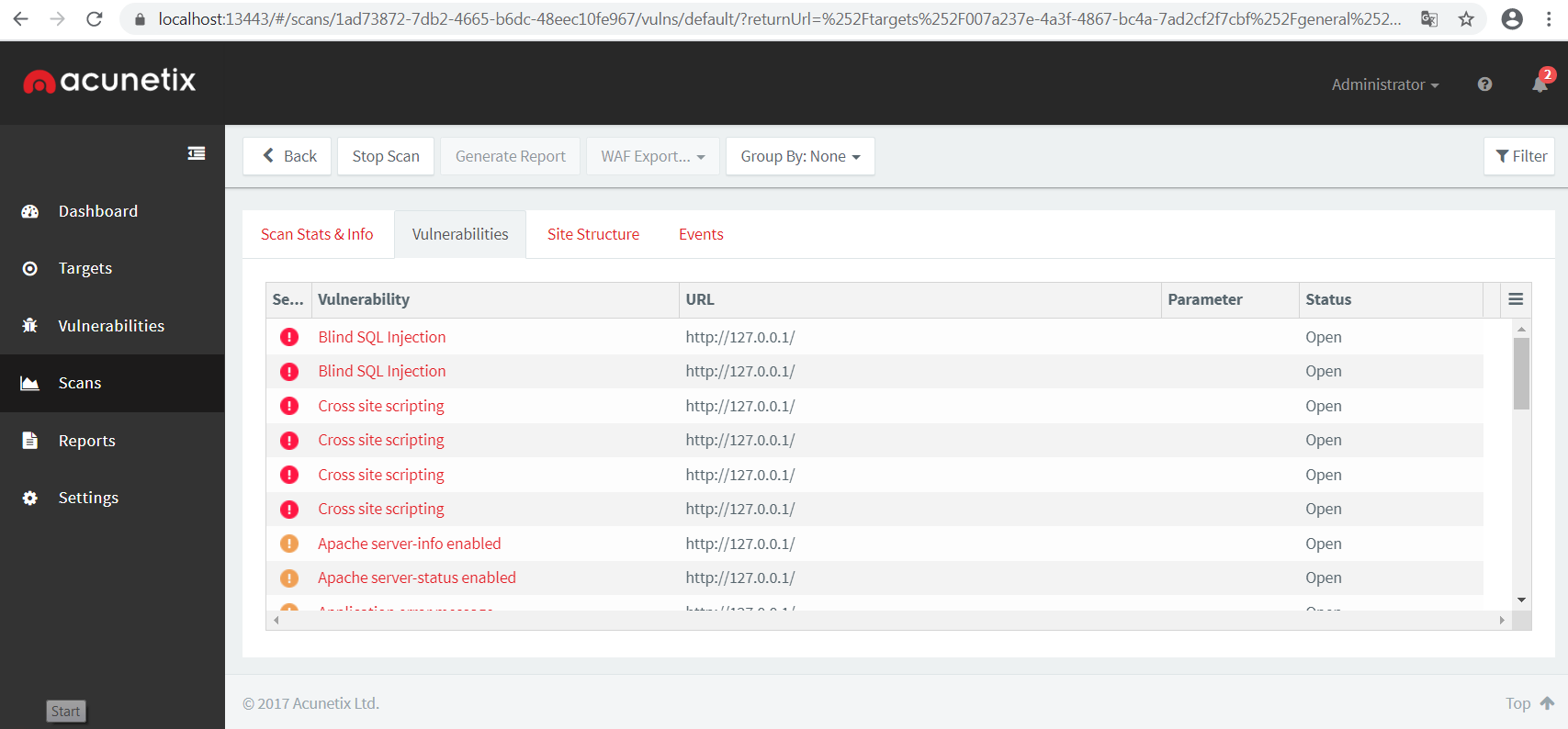


* $username và $password được lấy luôn từ người dùng nhập vào không qua filter để lọc các ký tự đặc biệt.
* $query không sử dụng parameter sử dụng cộng chuỗi.

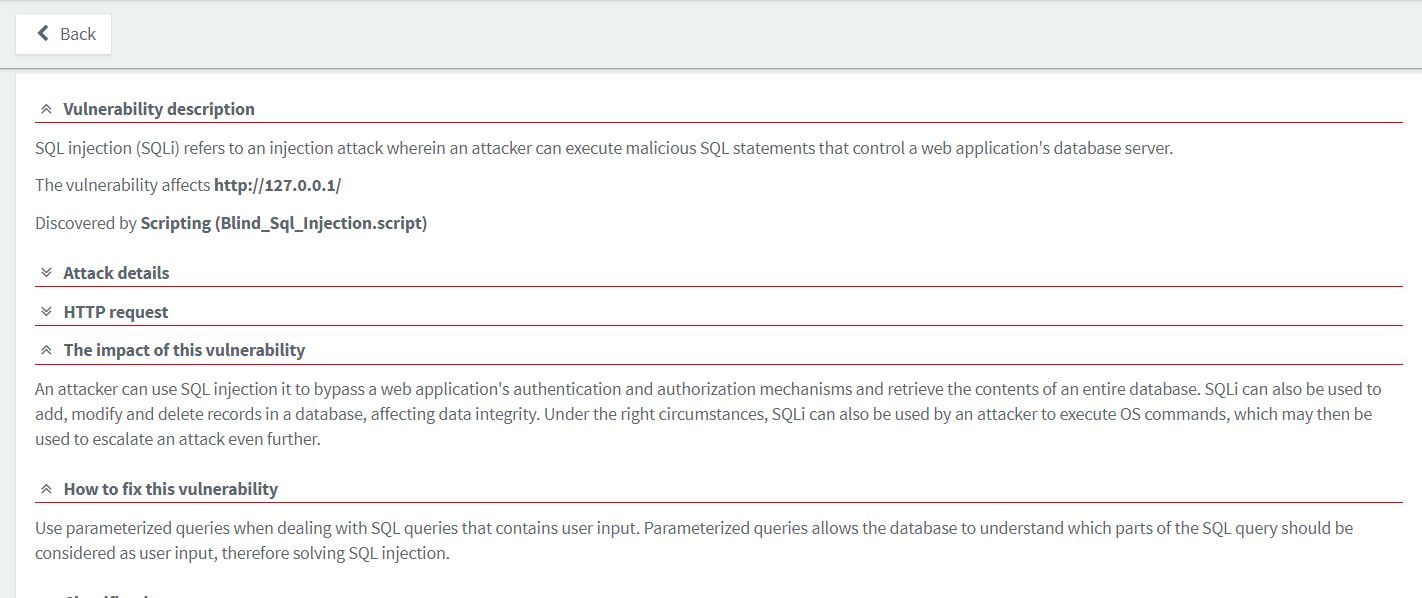
Truy cập http://127.0.0.1/test12/



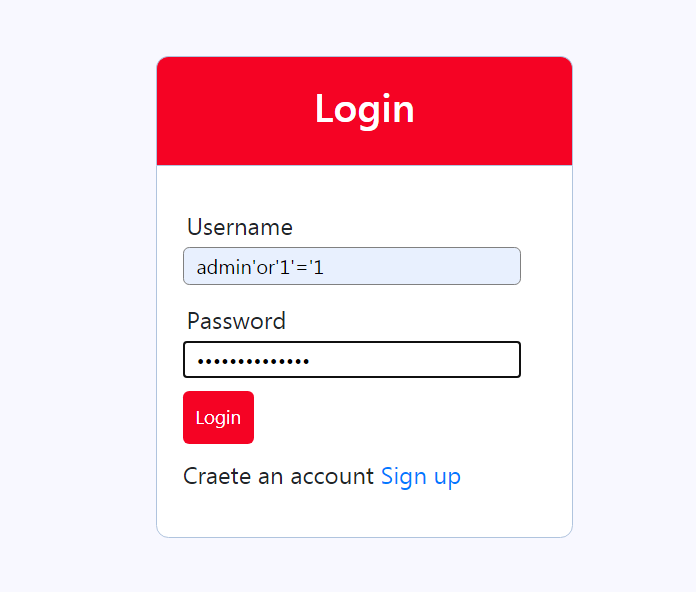
Cài đặt và sử dụng công cụ Acunetix – Scan thực hiện scan trên cùng máy ảo thực hiện scan http://127.0.0.1/test12/ được kết quả:



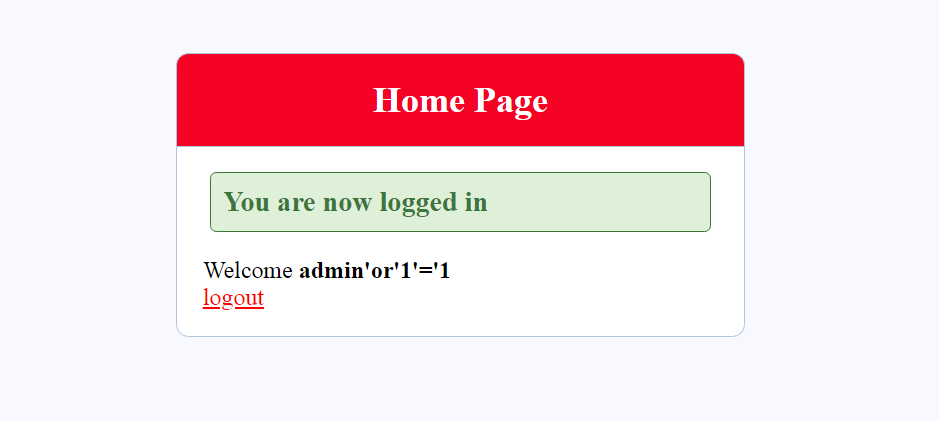
Tool đã quét được các lỗ hổng nghiêm trọng đó có SQL Injection với mô tả lỗ hổng và cách khắc phục:



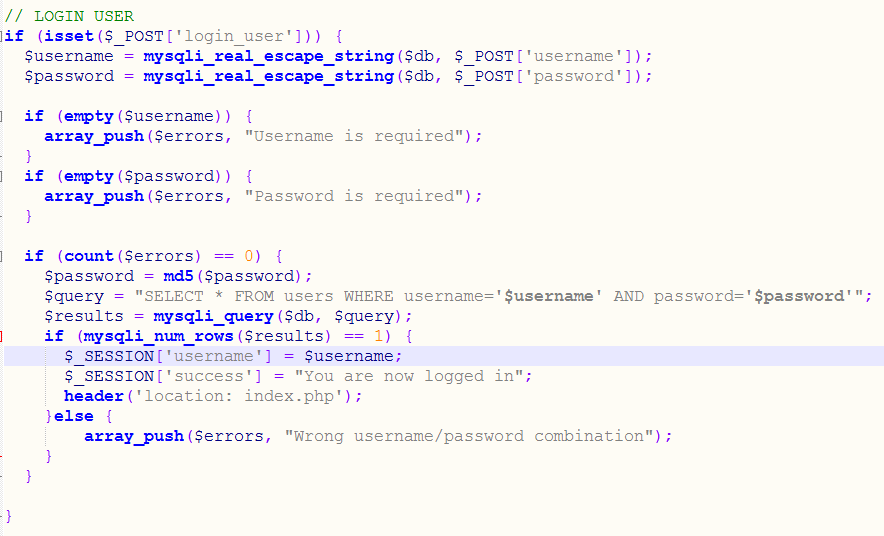
Thử khai thác lỗ hổng trên với from login nhập vào chuỗi admin'or'1'='1 ở cả username và password:



Kết quả có thể đăng nhập thành công:

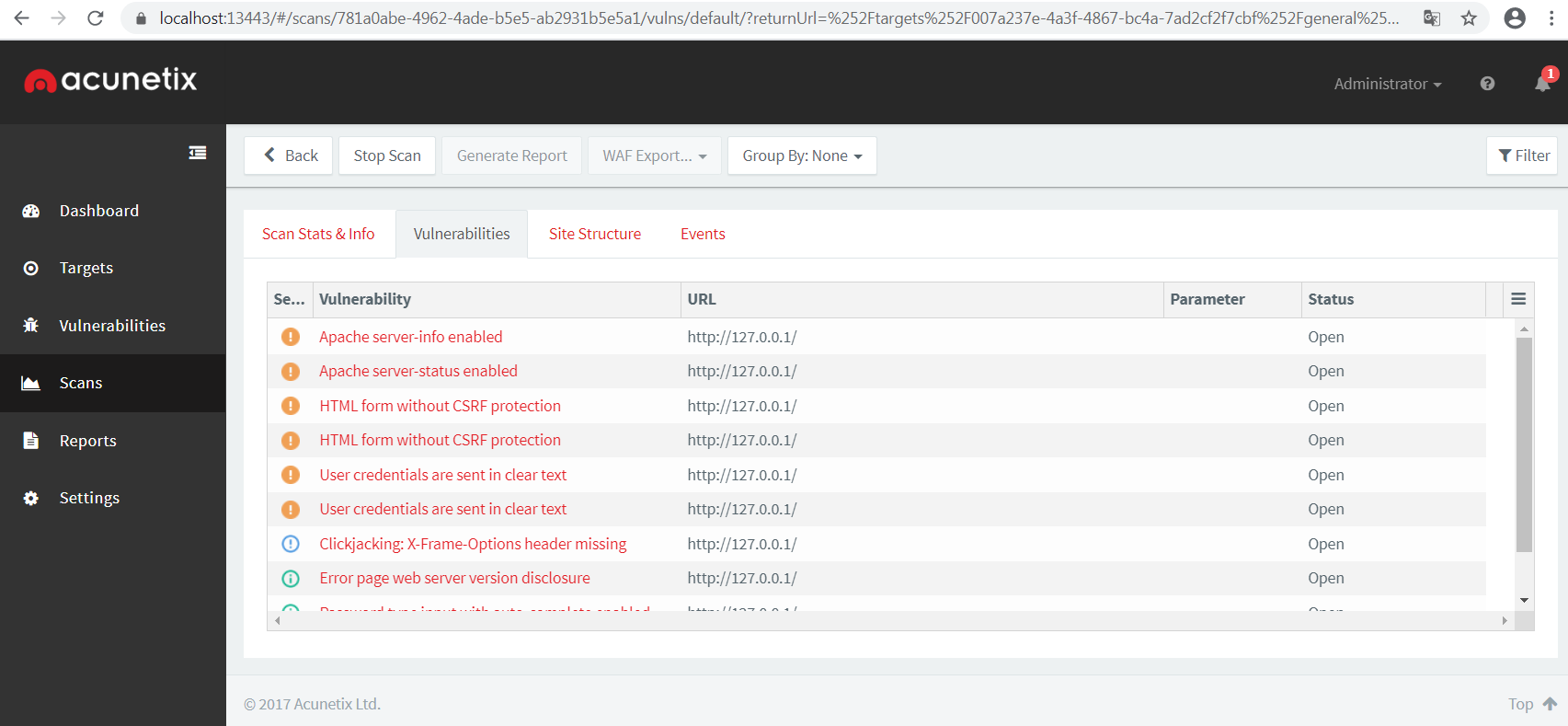


Cách khắc phục chống khai thác lỗ hổng SQL Injection cho web trên:

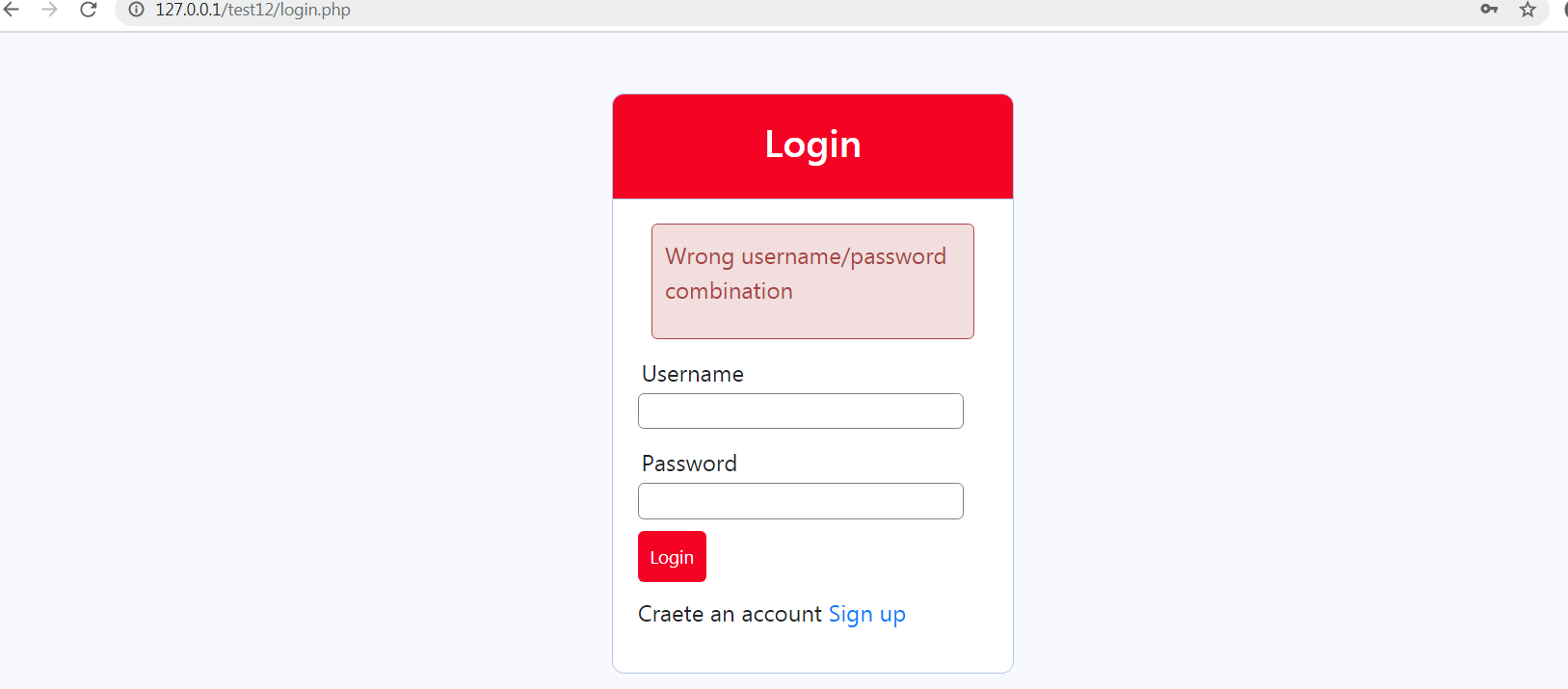


* **Dữ liệu đó có chứa các kí tự đặc biệt (ví dụ như dấu ‘ , ” , /, …) làm cho câu query của chúng ta bị lỗi cú pháp,**  ta cần phải có hàm để xử lý các biến trước khi đưa vào câu query.Ở đây ta sử dụng **mysqli\_real\_escape\_string** hàm xử lý kí tự đặc biệt của hệ quản trị cơ sử dữ liệu.
* Sử dụng mã hoá password **md5($password).**

Sử dụng công cụ Acunetix – Scan quét lại lỗ hổng trên http://127.0.0.1/test12/ có kết quả như sau:

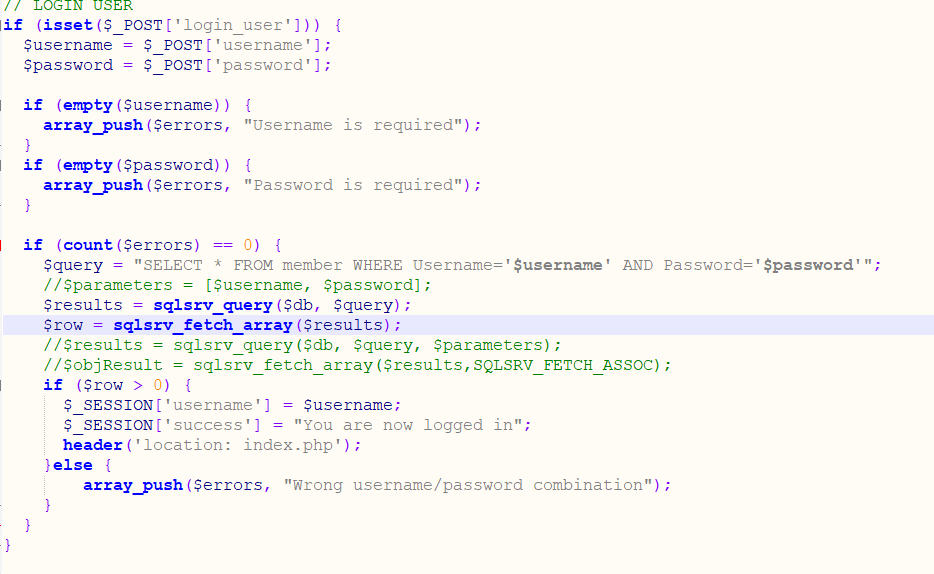


Các lỗ hổng nghiêm trọng về SQL Injection đã không còn thử khai thác lại nhập vào chuỗi admin'or'1'='1 ở from login kết quả:



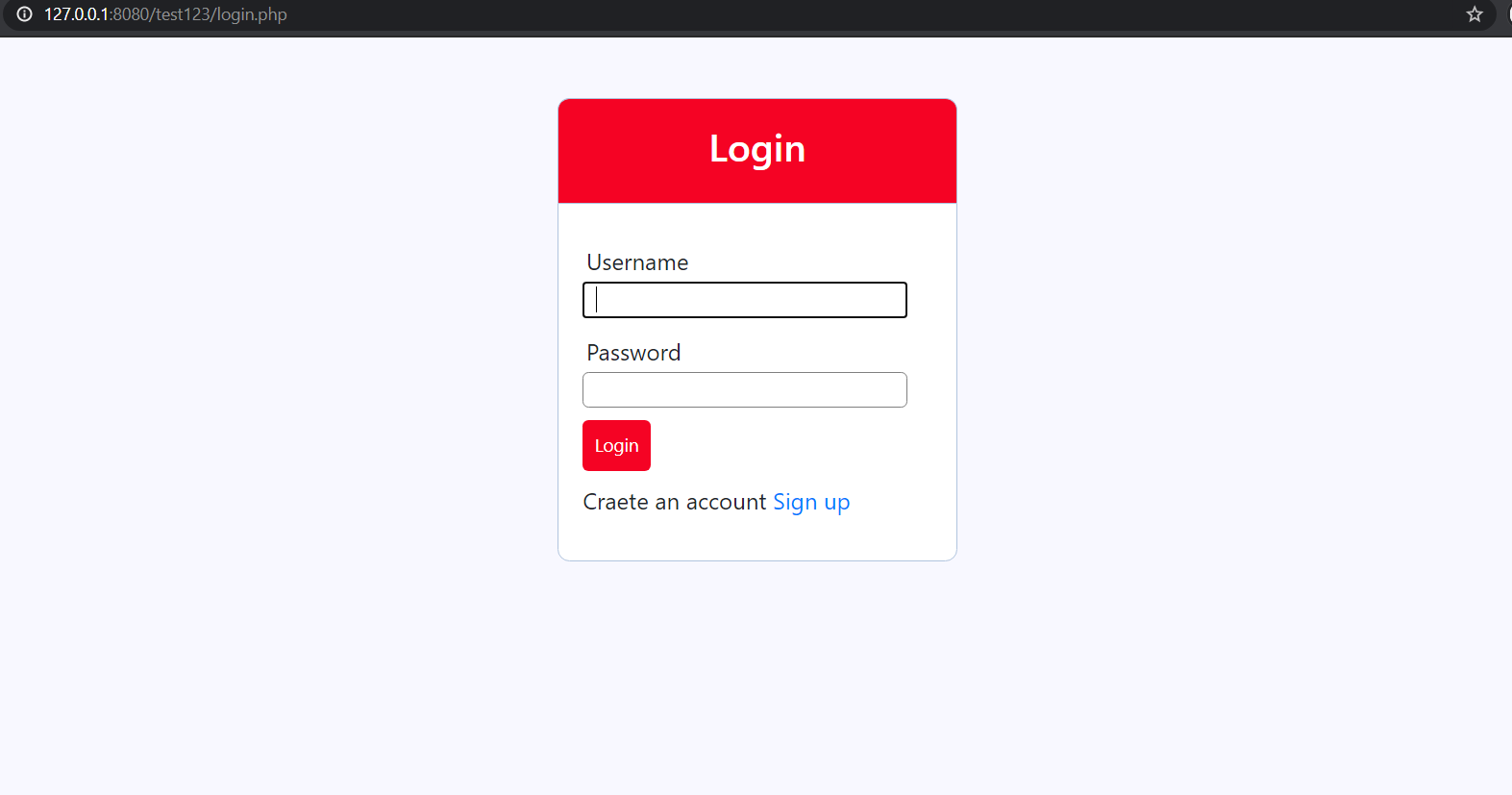
1. **Microsoft SQL Server**

Clone code trên github về cài đặt trên máy win 10 với Microsoft SQL Server và XAMPP. Sửa lại query SQL ta có đoạn code như sau:

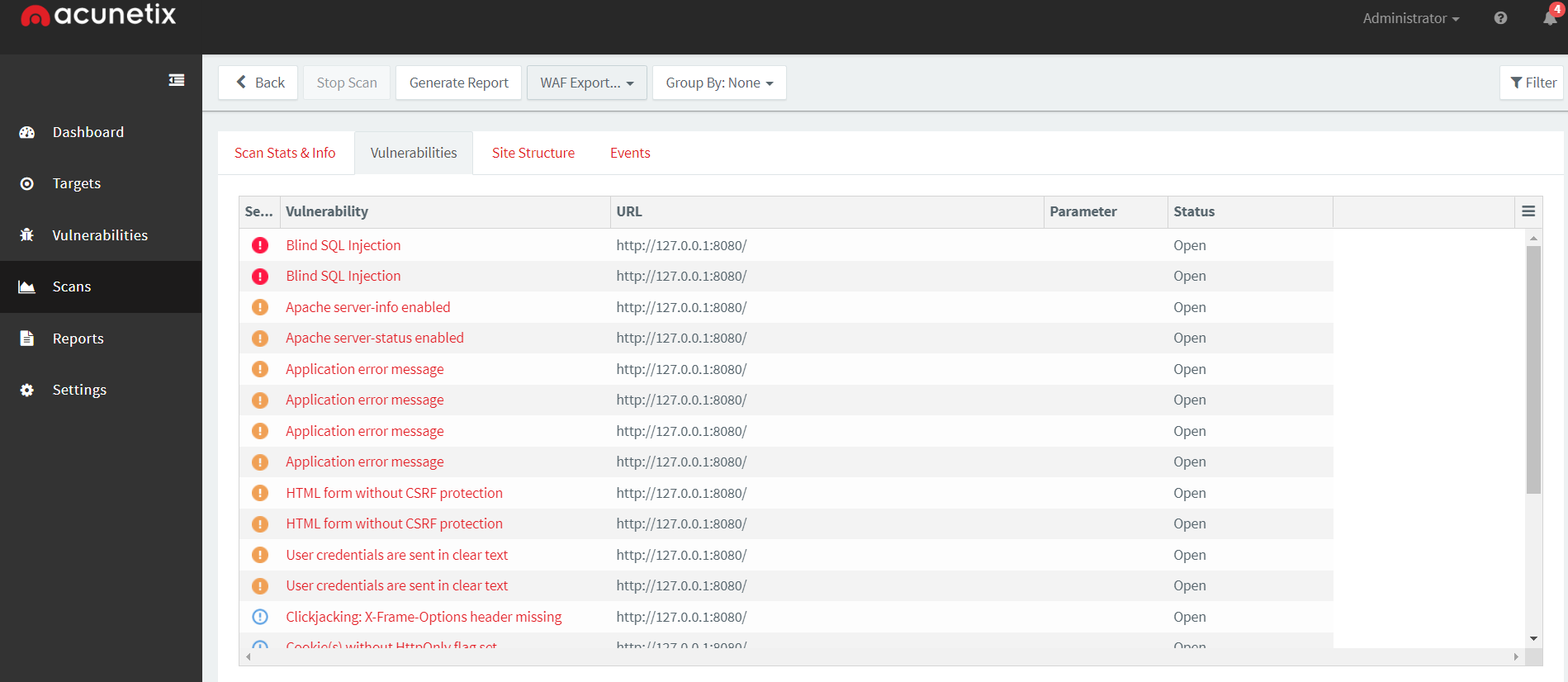


* $username và $password được lấy luôn từ người dùng nhập vào không qua filter để lọc các ký tự đặc biệt.
* $query không sử dụng parameter mà sử dụng cộng chuỗi.

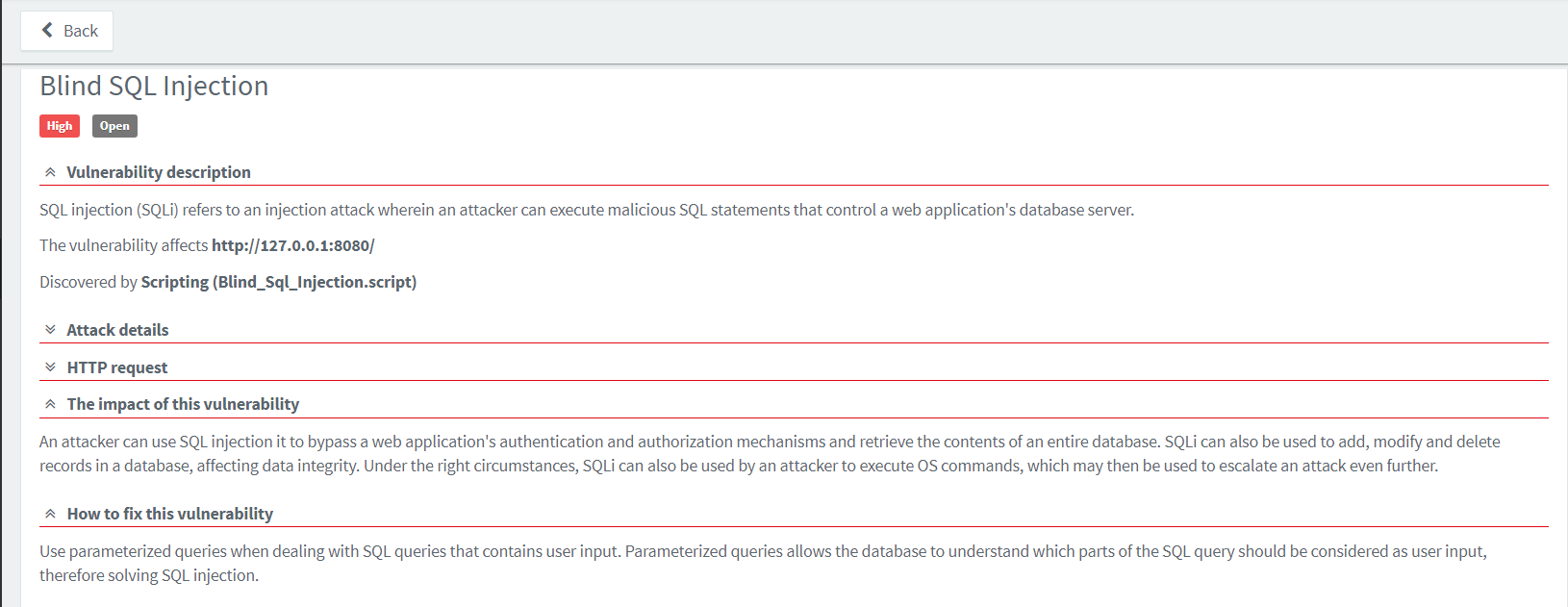
Truy cập http://127.0.0.1:8080/test123/



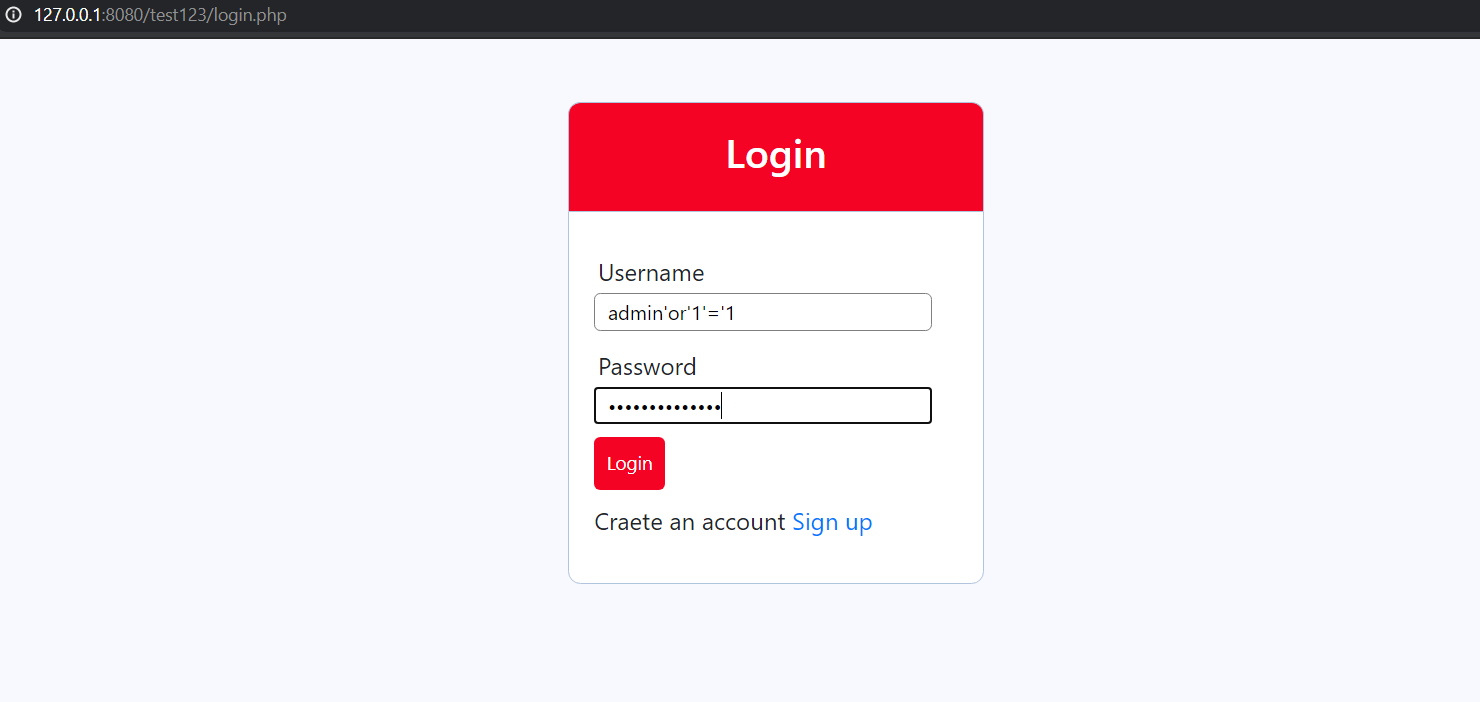
Cài đặt và sử dụng công cụ Acunetix – Scan thực hiện scan trên cùng máy ảo thực hiện scan http://127.0.0.1:8080/test123/ được kết quả:



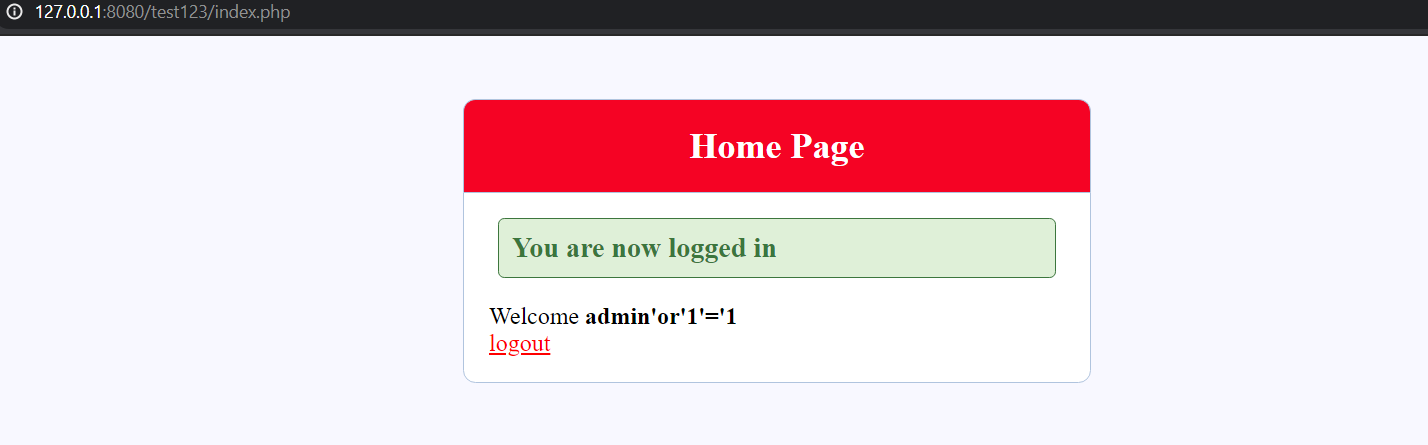
Tool đã quét được các lỗ hổng nghiêm trọng đó có SQL Injection với mô tả lỗ hổng và cách khắc phục:



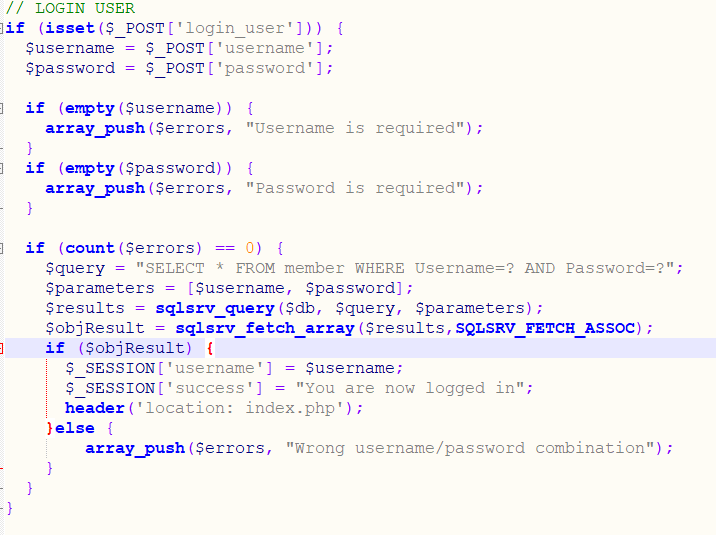
Thử khai thác lỗ hổng trên với from login nhập vào chuỗi admin'or'1'='1 ở cả username và password:



Kết quả có thể đăng nhập thành công:

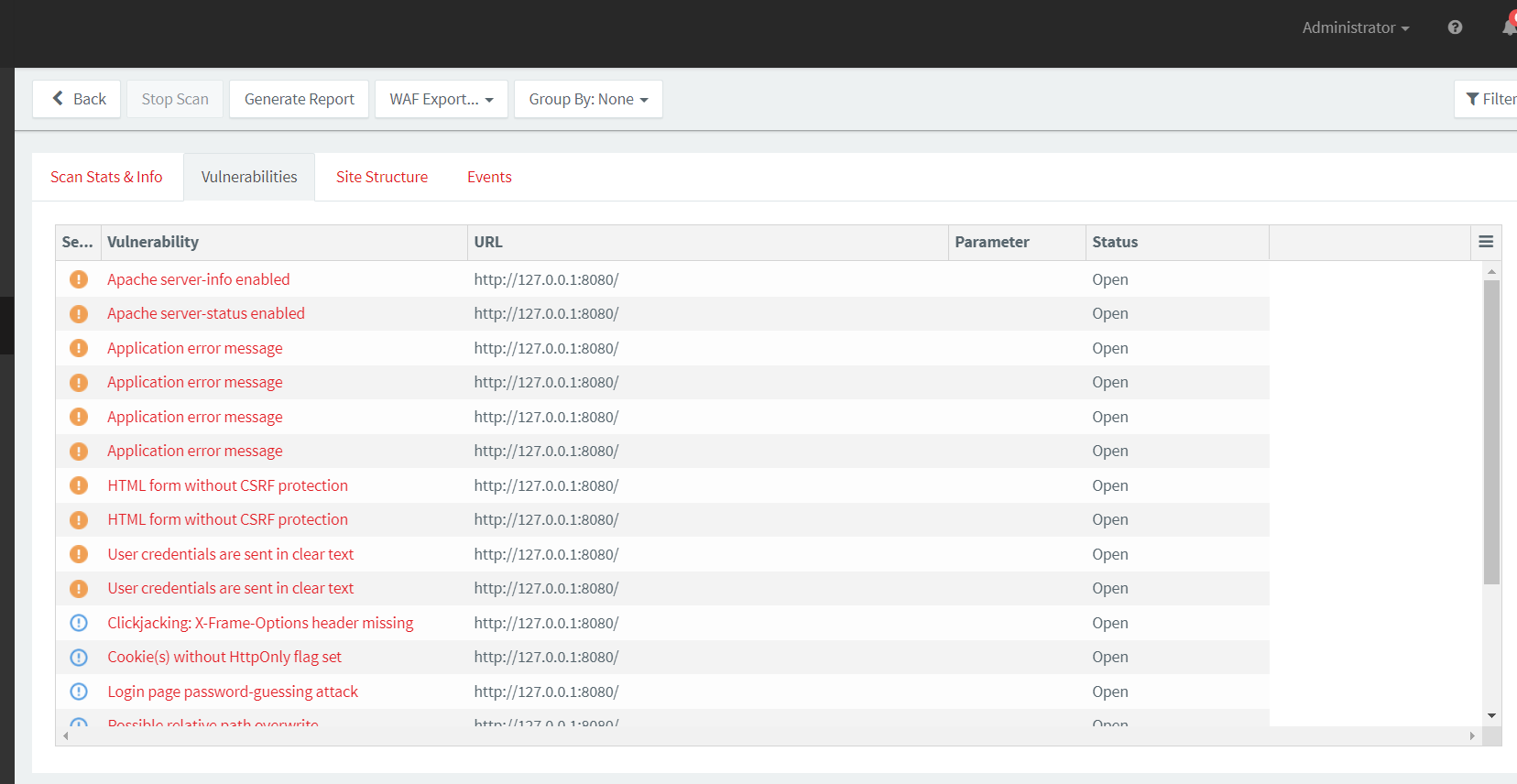


Cách khắc phục chống khai thác lỗ hổng SQL Injection cho web trên:



* Sử dụng parameters cho câu lệnh query**.**

Sử dụng công cụ Acunetix – Scan quét lại lỗ hổng trên http://127.0.0.1:8080/test123/ có kết quả như sau:



Các lỗ hổng nghiêm trọng về SQL Injection đã không còn thử khai thác lại nhập vào chuỗi admin'or'1'='1 ở from login kết quả:

