**SQL injection là:** kỹ thuật tấn công mà kẻ tấn công nhập các câu lệnh SQL độc hại hoặc gian lận vào các trường dữ liệu của các ứng dụng web có liên kết với cơ sở dữ liệu.

**Mục tiêu :** lợi dụng lỗ hổng trong việc xử lý dữ liệu đầu vào từ người dùng để thực hiện các hoạt động không được ủy quyền trong cơ sở dữ liệu hoặc gây ra các vấn đề bảo mật

**Tìm hiểu các câu truy vấn thường dùng**( select\*from), các cú pháp thường gặp( comment, substring, String Concatenation, time delay)

**Các cách phòng chống:**

* kiểm tra và lọc dữ liệu đầu vào
* Sử dụng Prepared Statements hoặc Parameterized Queries
* Hạn chế quyền truy cập của người dùng vào cơ sở dữ liệu

**5 cách tấn công cơ bản**

* **Union-base:** thuộc vào loại In-band SQLi. Kết quả truy vấn trả về dưới dạng phản hồi của trang web, có thể dùng toán tử UNION để truy xuất dữ liệu từ các bảng khác trong cơ sở dữ liệu. Nó cho phép thực hiện nhiều câu lệnh SELECT để truy vấn. Kẻ tấn công sẽ chèn câu truy vấn UNION SELECT vào câu truy vấn ban đầu để thực hiện truy vấn bổ sung tới các bảng khác trong cơ sở dữ liệu.
* Điều kiện thực hiện: cùng số cột, cùng kiểu dữ liệu
* Thực hành bằng BurpSuite
* Chèn giá trị NULL để xác định số cột có trong bảng
* Kiểm tra kiểu dữ liệu của từng cột
* Lấy username và password của người dùng
* Thực hành bằng Python
* Tìm số cột của bảng bằng cách dùng ORDER BY
* Kiểm tra kiểu dữ liệu của từng cột
* Lấy username và password của người dùng
* **Error-base:**thuộc inband SQLi. Kẻ tấn công lợi dụng việc server trả về một thông báo lỗi quá chi tiết (Có thể bao gồm vị trí lỗi, lỗi như nào,...) qua đó có khả năng trích xuất được những thông tin nhạy cảm qua thông báo lỗi đó. Ngoài ra, error-based còn được kết hợp với boolean-based để đối phó với những trường hợp thông báo lỗi trả về chỉ là một thông báo lỗi tổng quát chứ không chứa chi tiết thông tin
  + Thực hành bằng burpsuite:
    - Kiểm tra xem nếu xảy ra lỗi web có trả về một thông báo lỗi chi tiết hay không bằng cách thêm bất kì ký tự đặc biệt hay truy vấn nào gây ra lỗi
    - Thu nhỏ dần phạm vi gây lỗi để chi tiết được trả về trong thông báo lỗi sẽ là vị trí mà ta cần trích xuất dữ liệu (Nếu cần lấy username thì thực hiện các truy vấn gây lỗi cho phần bảng username,....)
* **Boolean-base**: Là một dạng blind SQL injection. Kẻ tấn công thường sử dụng các biểu thức boolean như "AND", "OR", "NOT" để kiểm tra các điều kiện khác nhau và thu thập thông tin từ cơ sở dữ liệu và phân tích sự khác biệt trong phản hồi giữa các câu TRUE và FALSE. Từ đó dự đoán trang web có thể bị tấn công hay không.
* Thực hành bằng burpsuite
* Kiểm tra tham số đưa vào có chứa lỗ hổng blind SQL hay không: gửi request Union, thêm payload ‘And 1=1’
* Kiểm tra sự tồn tại của bảng users trong CSDL của ứng dụng web
* Kiểm tra sự tồn tại của quản trị viên trong CSDL và username
* Tìm độ dài của mật khẩu quản trị viên
* Tìm giá trị chính xác của mật khẩu quản trị viên
* Thực hành bằng python: Hỗ trợ trích xuất mật khẩu của quản trị viên thông qua lỗ hổng SQL injection. Nó thực hiện các bước sau:
  + Thiết lập các biến và proxies cho việc debug.
  + Lặp qua các ký tự ASCII và chữ số để thử từng ký tự một.
  + Tạo cookies chứa chuỗi SQL injection tương ứng với từng ký tự và gửi yêu cầu HTTP GET đến URL được cung cấp.
  + Kiểm tra xem gợi ý đã được tìm thấy trong phản hồi hay không. Nếu có, ký tự đó được thêm vào mật khẩu và tiếp tục tìm ký tự tiếp theo.

- **Out-of-band**: sử dụng khi phản hồi của Server không ổn định hoặc ứng dụng thực hiện truy vấn SQL bất đồng bộ, dẫn đến các dạng tấn công như boolean-based, time-based, error-based,... trở nên không còn phù hợp. Kẻ tấn công sẽ sử dụng các câu truy vấn với mục tiêu điều hướng DNS request, HTTP request,... trong request có thông tin cần lấy ở Database Server, các request được điều hướng đi qua một server do kẻ tấn công làm chủ và thông qua việc xem lưu lượng đi qua kẻ tấn công có thể dễ dàng lấy được thông tin

- Kiểm thử bằng burpsuite:

* Kiểm tra kết nối tới Burp Collaborator, bằng cách tạo một câu truy vấn với chức năng tạo một DNS request tới Burp Collaborator mà ta sở hữu
* Cũng với câu truy vấn trên thay vì truy vấn tới Burp Collaborator ta sẽ truy vấn tới một subdomain của nó với tên subdomain sẽ là dữ liệu mà ta cần lấy, sử dụng ký tự nối chuỗi để nối dữ liệu vào
* Sau khi gửi request đi ta sẽ nhận được một request đi qua Burp Collaborator, với định dạng thông tin cần lấy nằm ở ngay trước đường dẫn cơ bản của ta
* **Time-based Sql**:là kiểu tấn công SQL Injection dựa vào việc gửi một truy vấn SQL đến cơ sở dữ liệu, buộc cơ sở dữ liệu phải đợi một khoảng thời gian nhất định (tính bằng giây) trước khi phản hồi. Thời gian phản hồi sẽ cho biết cho kẻ tấn công biết kết quả của truy vấn là TRUE hay FALSE
* Kiểm thử bằng burpsuite:
* Sử dụng các truy vấn SQL chậm để kiểm tra và khai thác lỗ hổng ứng dụng web
* Khi hacker gửi một truy vấn SQL độc hại, nó sẽ tạo ra một trễ thời gian trong quá trình xử lý của cơ sở dữ liệu, giúp hacked xác định thông tin bí mật từ cơ sở dữ liệu
* Hacker có thể thực hiện tấn công bằng cách sử dụng các kỹ thuật như IF, SLEEP, hoặc BENCHMARK để tạo trễ thời gian trong câu lệnh truy vấn.