SQL Injection

SQL 삽입(SQL Injection)은 웹 애플리케이션의 보안 취약점을 이용하여 공격자가 악성 SQL 코드를 삽입하여 데이터베이스에 접근하거나 조작하는 공격이다. 이를 통해 공격자는 데이터베이스에 저장된 정보를 탈취하거나 수정할 수 있으며, 가장 흔한 웹 해킹 공격 중 하나이다.

SQL 삽입 공격은 주로 웹 애플리케이션의 입력 폼, URL 매개 변수, 쿠키 값 등 사용자 입력을 처리하는 곳에서 발생할 수 있다. 공격자는 여러 기법을 사용하여 삽입된 SQL 코드를 실행시키며, 이를 통해 데이터베이스에 접근하거나 조작한다.

SQL 삽입 공격 작동 원리

1. **입력 처리 과정**: 웹 애플리케이션은 사용자 입력을 받아서 쿼리문을 생성하거나 처리한다.
2. **악성 입력 삽입**: 공격자는 입력 폼 등을 통해 악성 SQL 코드를 삽입한다. 예를 들어, 로그인 폼에 `username`과 `password` 대신에 `' OR '1'='1`과 같은 코드를 입력할 수 있다.
3. **SQL 조작**: 공격자가 삽입한 악성 코드가 서버에서 쿼리문으로 해석되면, 데이터베이스는 조작된 SQL 문을 실행하게 된다.
4. **결과 반환**: 조작된 SQL 쿼리문의 결과나 데이터베이스에서 추출한 정보가 공격자에게 반환된다.

SQL 삽입 공격 방어 대책

1. **입력 검증 및 이스케이프**: 사용자 입력값을 검증하고 쿼리문을 생성할 때 이스케이프 함수를 사용하여 악성 코드를 무력화해야 한다.

2. **매개 변수화된 쿼리**: 매개 변수화된 쿼리(prepared statement)를 사용하여 사용자 입력을 쿼리에 직접 포함하지 않고 처리해야 한다.
3. **접근 권한 제한**: 데이터베이스 사용자에게 최소한의 권한만 부여하여 악성 공격 시 피해를 최소화해야 한다.
4. **보안 커뮤니티 리소스 활용**: OWASP 등 보안 커뮤니티의 가이드와 도구를 활용하여 보안 취약점을 탐지하고 대응하는 방법을 탐구한다.
5. **정기적인 보안 테스트**: 웹 애플리케이션을 정기적으로 보안 테스트하여 취약점을 탐지하고 수정해야 한다.