SQL Injection

사용자에게 입력을 받는 칸에 개발자가 의도한 정상 데이터가 아닌, 데이터베이스에서 사용되는 질의문을 삽입하여 데이터베이스의 정보를 검색, 추가하는 등 조작을 가하거나 인증을 우회.

- 인증 우회: 1' or 1 #

  # 이하는 주석처리 되어 인증을 우회할 수 있다.

- Union: union은 두 개의 질의문에 대한 결과를 통합하여 보여줌. 하나의 질의문에 데이터베이스를 조작하는 명령을 입력하여 공격.
- Blind: 알아내고 싶은 정보와 참인 것을 알고 있는 정보를 AND를 이용해 하나의 질의문으로 입력. 알아내고 싶은 데이터의 값을 바꾸어가며 유추.

탐지 방법

사용자의 입력값이 패킷으로 전송될 테니 패킷의 내용 중 SELECT, OR 등 데이터베이스 쿼리문에 사용되는 키워드가 포함되어 있는지 여부를 파악하는 것으로 탐지.

XSS

게시판 또는 url을 통해서 사이트에 악성 스크립트를 삽입하고, 그를 통해 데이터를 탈취해 피해자의 세션을 하이제킹할 수 있음.

- Stored: 악성 스크립트를 서버에 저장. 주로 게시판을 이용한다. 사용자가 게시물에 접근하면 악성 스크립트가 실행되면서 공격당함. 클라이언트 단에서 실행됨.
- Reflected: 스크립트가 저장되지 않고 사용자의 브라우저에서 바로 실행됨.

탐지 방법

Stored의 경우 공격자가 게시판 등을 통해 스크립트를 서버에 저장하는 방식이기 때문에 스크립트를 포함한 패킷이 전송될 것. 해당 패킷을 잡는 것으로 탐지.

Reflected의 경우 쿠키 정보를 요청하는 내용의 패킷이 전송이 되거나, 쿠키 정보를 담은 패킷이 아예 엉뚱한 서버(공격자의 서버)로 전송되는 것을 탐지.

CSRF

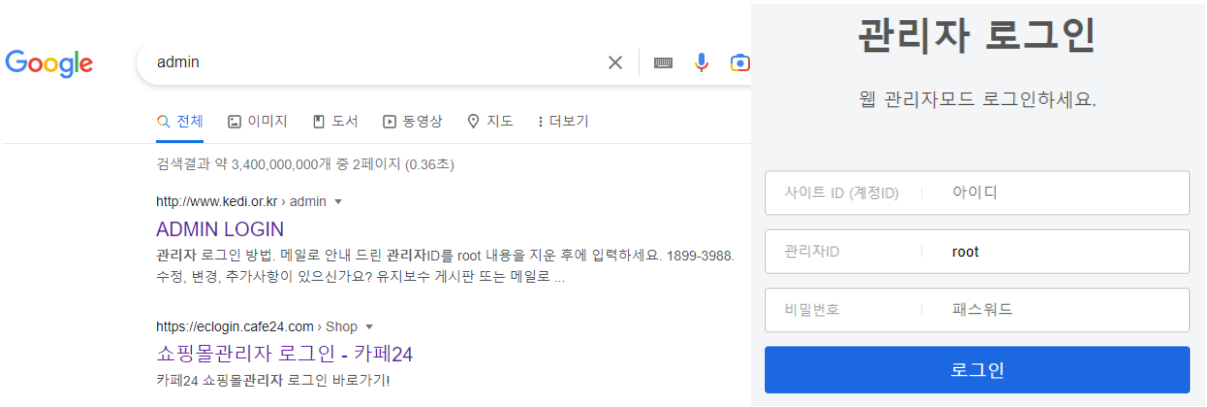
사이트에 로그인한 사용자가 자신의 의도와는 무관하게 공격자가 의도한 행위를 하게 하는 공격. 요청을 위조한 url을 사용자가 클릭하면 악성 스크립트가 서버에 요청됨.

탐지 방법

비밀번호를 변경하는 요청을 담은 url을 통해 공격을 한다고 가정. 새로운 비밀번호에 대한 정보를 url에 표시하는 get 방식의 요청이 들어온다면 해당 패킷을 탐지. 비밀번호 등 개인정보는 post 방식으로 전달하는 것이 보통이라고 알고 있기 때문에 이런 방법으로 탐지할 수 있을 것 같음.

취약한 관리자 페이지

아래 그림과 같이 검색만 해봐도 취약한 페이지가 발견됨.



탐지 방법

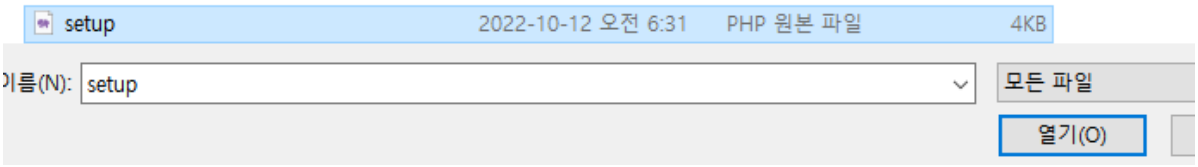
/admin을 여러 경로로 검색하는 등 디렉토리 경로 추측을 통해 관리자 페이지를 찾으려는 시도를 탐지. 관리자 페이지에 아이디와 비밀번호를 바꿔가며 여러 번 로그인을 시도하는 경우를 탐지.

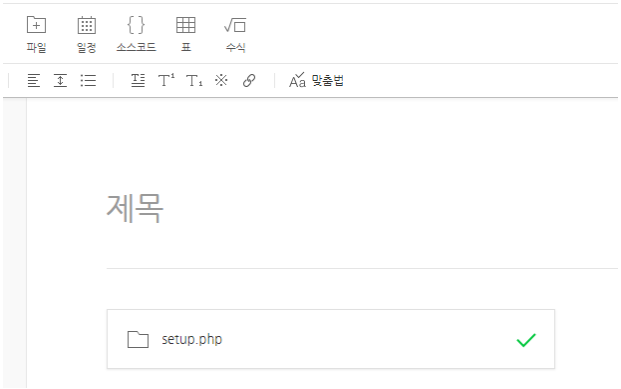
차단 방법

관리자 페이지에는 허용된 일부 ip만 접근이 가능하도록 설정하기.

파일 업로드 취약점

아래 그림과 같이 웹쉘 파일을 업로드 할 수 있음. 실행 되는지는 못 해봤음.





탐지 방법

공격자가 사이트에 웹쉘을 업로드하고, 그 웹쉘을 실행하기 위해 웹쉘이 저장된 디렉토리 경로를 찾는 과정의 패킷을 탐지할 수 있을 거 같음.

차단 방법

사용자가 업로드한 파일에 대한 패킷을 확인하여 exec 등 시스템에 실행을 요청하는 명령이 포함되어 있다면 차단. 서버 자원에는 한정된 일부 ip만 접근 가능하도록 제한하기.