XSS

#Cross_Site_Scripting

목차

XSS

Cross Site Scripting

- 관리자가 아닌 권한이 없는 사용자가 웹 사이트에 스크립트를 삽입하는 공격 기법

- 사용자의 권한 획득
- 사용자의 페이지 변조
- 제한된 웹 사이트 접근
- 브라우저 원격제어

XSS 공격 기법

1

스크립트 태그

<script>alert('XSS');</script>

2

링크 태그

XSS

3

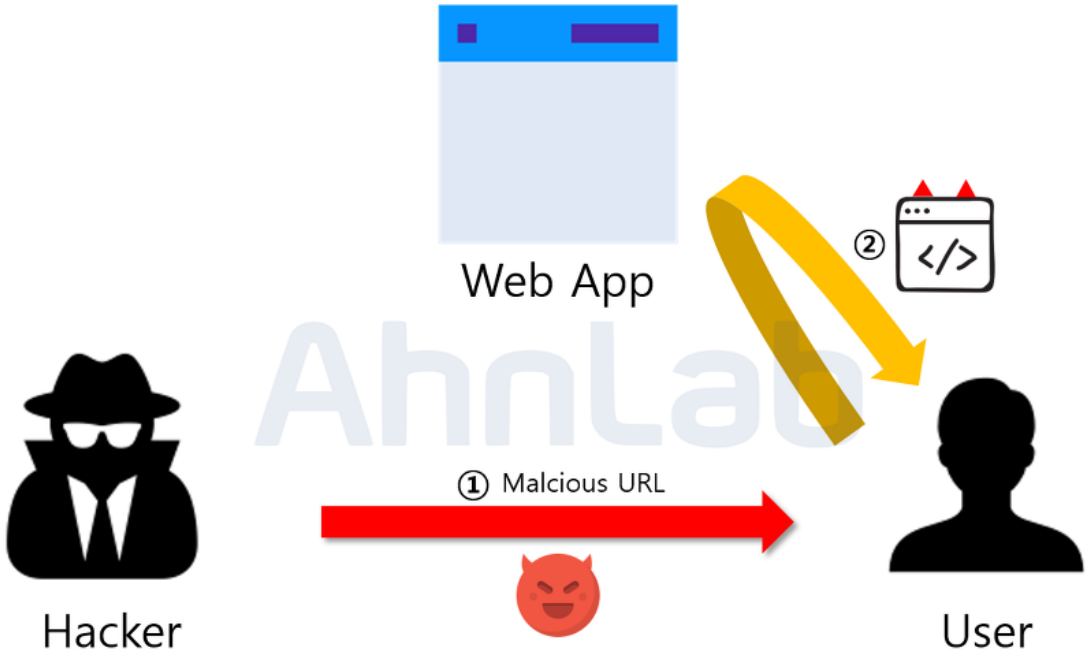
이벤트 속성

<img src="#" onerror="alert('XSS')">

Stored XSS
저장형 XSS 공격

- 악성 스크립트 삽입 → 데이터베이스에 저장 → 사용자 열람 → 악성 스크립트 작동
- 웹 게시판
- 불특정 다수에게 공격 가능 → 수많은 피해

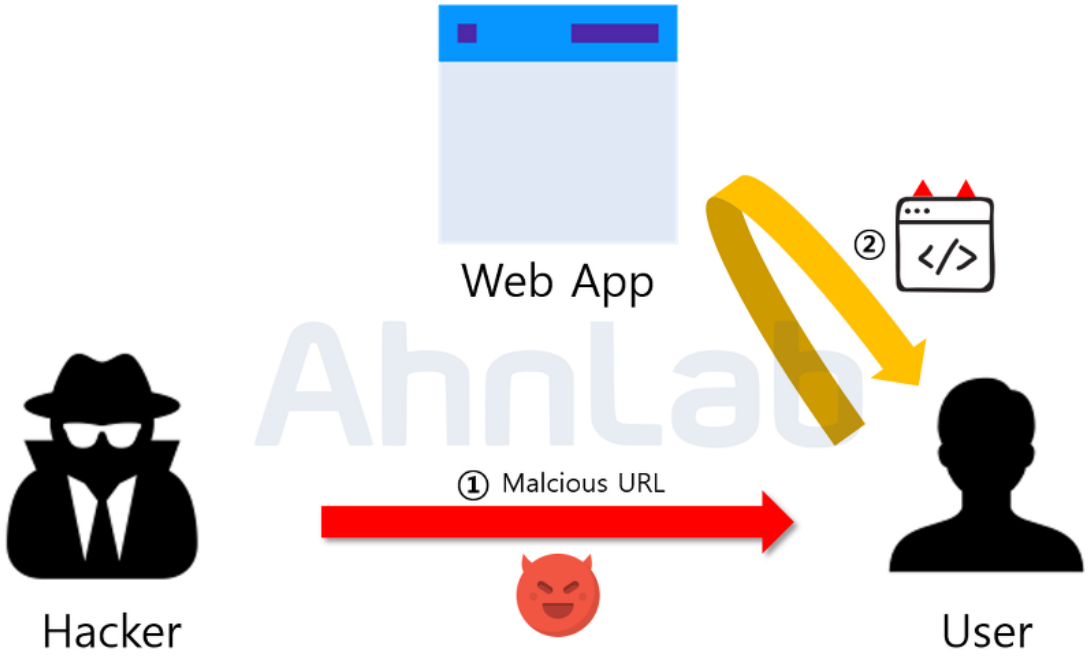
Reflected XSS
반사형 XSS 공격



- 웹 애플리케이션의 지정된 변수를 이용할 때 발생하는 취약점을 이용한 공격
- 사용자의 요청 데이터가 서버의 응답에 포함되는 과정에서 HTML 등의 악성 스크립트가 그대로 출력되어 발생
- 악성 URL을 누르도록 유도하여 공격

Reflected XSS

반사형 XSS 공격 단계



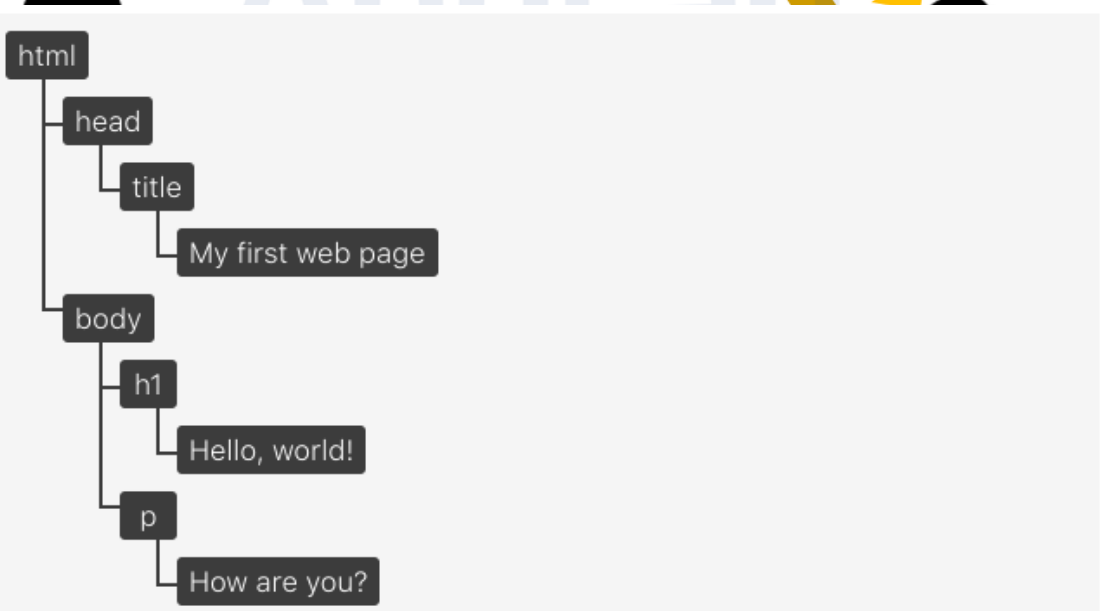
1. 공격자는 사이트에 취약점을 발견
2. 공격용 악성 URL 생성
3. 공격자는 이 URL을 포함하여 배포
4. 피해자가 URL을 클릭하면, 바로 공격 스크립트가 사이트와 관련된 정보를 공격자에게 전송

Reflected XSS
반사형 XSS 공격

```
<html>

<body>

<div id="pageTitleTxt">

<h2><span class="highlight">Search Results</span><br />

Search: "<script>alert(document.cookie)</script>"</h2>

</body>

</html>
```

DOM based XSS
DOM 기반 XSS 공격

```
<!doctype html>
<html lang="en">
 <head>
   <title>My first web page</title>
  </head>
 <body>
    <h1>Hello, world!</h1>
    <p>How are you?</p>
  </body>
</html>
```



- DOM 객체를 실행할 때 URL 등에 포함된 악성 스크립트가 동작하는 방식
- 정상적인 스크립트가 동작하면서 DOM 객체를 실행할 때 URL 등에 포함된 악성 스크립트가 동작
- 브라우저에서 발생

DOM based XSS

DOM 기반 XSS 공격

```
<HTML>

<TITLE>Welcome!</TITLE>

Hi

<script>

 var pos=document.URL.indexOf("name=")+5;

 document.write(document.URL.substring(pos,document.URL.length));

</script>

<br>

Welcome to our system
This demo borrowed from http://www.webappsec.org/projects/articles/071105.shtml

</HTML>
```

http://www.server.com/page.html?name=<script>alert(document.cookie)</script>

http://server/page.html?name=David#<script>alert(document.cookie)</script>

입출력 값 검증 및 무효화

HTML 엔티티
사용자가 입력한 특수 문자가
순순히 문서에 출력되고자 하는
일반 문자임을 구분하는 표기법

한 길이, 문자, 형식 및 사업적 규칙 유효성을 검사

기본적으로 <script> 태그를 사용하기 때문에 XSS공격을 차단하기 위해
태그 문자 (<,>)등 위험한 문자 입력 시 HTML entity로 필터링
서버에서 브라우저로 전송 시 문자를 인코딩

<script> → <script>

ASCII 문자	참조 문자	ASCII 문자	참조 문자
&	&amp;	"	&quot;
<	&lt;	'	&#x27;
>	&gt;	/	&#x2F;
(	&#40;	)	&#41;

HTTPOnly Flag

document.cookie와 같은 자바스크립트로 쿠키를 조회하는 것을 막는 옵션

브라우저에서 HTTP Only가 설정된 쿠키를 조회할 수 없다

클라이언트 측의 스크립트가 보호된 쿠키에 접근할 때 위험을 완화시키는 역할

Content Security Policy(CSP)

웹 브라우저에서 사용하는 컨텐츠 기반의 보안 정책

페이지 설정으로 인해 자원 읽기 차단

Content-Security-Policy: default-src ‘self’ abcde.com

Content-Security-Policy: default-src 'self’ *.abcde.com; img-src *

X-XSS-Protection Header

공격자가 XSS 공격을 시도할 때 브라우저의 내장 XSS Filter를 통해
공격을 방지할 수 있는 헤더
→웹 브라우저에 내장된 XSS Filter를 활성화할 것인지를 설정

Request 값과 Response를 비교해 판단
→ Reflected XSS 공격을 막는 데에 적합한 방어 방법

X-XSS-Protection: 0
X-XSS-Protection: 1
X-XSS-Protection: 1; mode=block
X-XSS-Protection: 1; report=<reporting-uri>

Q&A