DOM Based Cross Site Scripting 취약점 분석

김세진

한양대학교 프로그래밍언어연구실

PLASSE RESEARCH

소개

- DOM Based Cross Site Scripting(DOM Based XSS)
  - 사용자 브라우저의 DOM(document object model)에 접근
  - 클라이언트 측에서 실행
  - 자바스크립트 기반 웹 애플리케이션을 위협하는 가장 위험한 취약점의 하나

DOM Based XSS 취약점을 유발하는 요소

- 비 신뢰성 외부 입력 (Untrusted Source)
  - document.location (and many of its properties),
  - document.URL,
  - document.URLUnencoded,
  - document.referrer
  - window.location (and many of its properties)
- 자바스크립트 실행 지점 (Critical Sink)
  - document.write()
  - document.writeln()
  - document.body.innerHtml
  - eval()

DOM Based XSS 취약점 예제

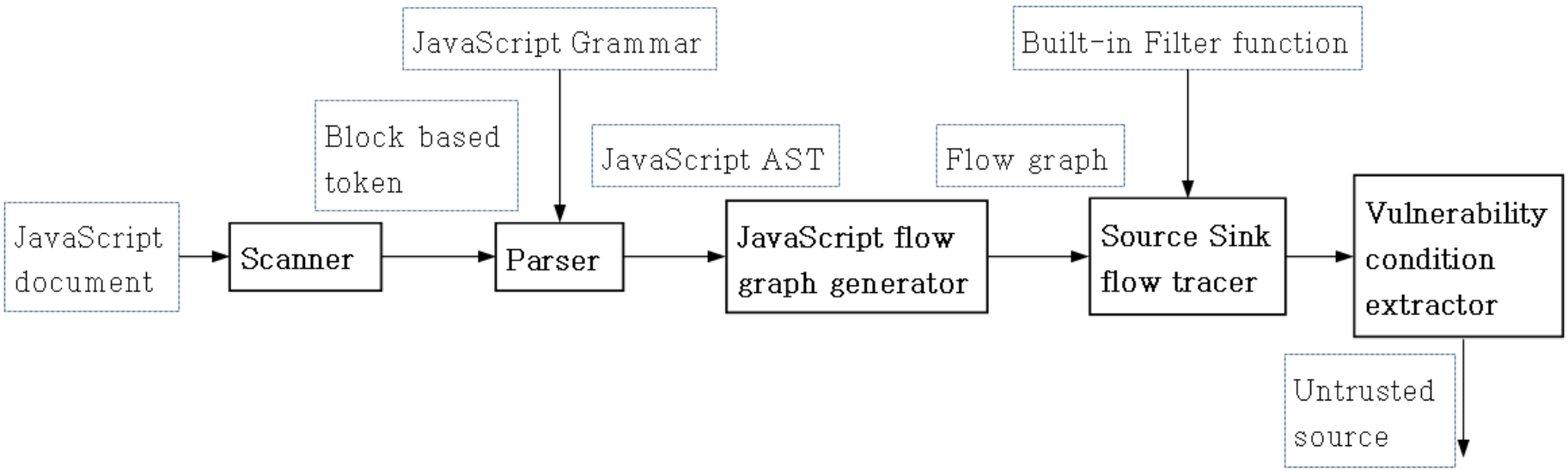
- 외부 입력:
  - 올바른 입력:
    http://www.vulnerable.site/welcome.html**?name=Joe**
  - 올바르지 않은 입력:
    http://www.vulnerable.site/welcome.html
    #name=<script>alert(document.cookie)</script>
- 내부 소스코드:

```
var position = document.URL.indexOf("name=")+5;
var name = document.URL.substring(position,
                                  document.URL.length);
document.write(name)
```

PLASSE RESEARCH

기존의 자바스크립트 정적 분석

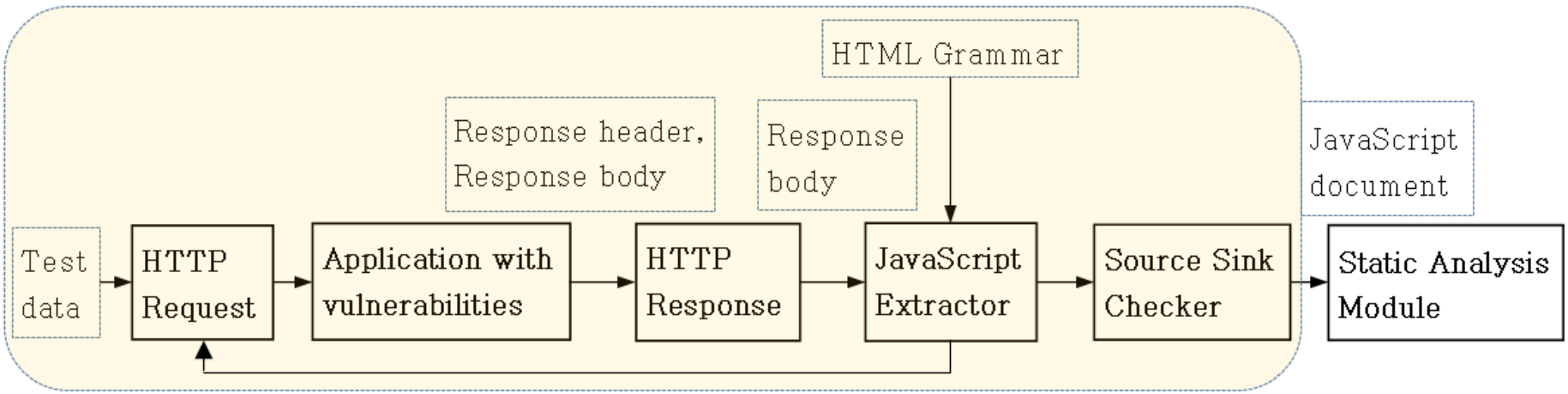
- 프로그램을 실행하지 않고 프로그램의 데이타 흐름을 기반으로 모든 실행 경로에서 취약점 존재를 분석



PLASSE RESEARCH

DOM Based XSS 취약점 동적 테스팅

- 웹 애플리케이션에 특정 요청을 보내 응답 메시지를 접수
- 응답 바디에서 자바스크립트 추출
- 비 신뢰성 외부 입력과 자바스크립트 실행지점 존재 확인



PLASSE RESEARCH

DOM Based XSS 취약점 동적 테스팅의 효과와 한계

- 효과
  - 실행 경로 다양성의 원인으로 생기는 미탐 없음
  - 연관된 외부 자바스크립트 파일을 동적으로 요청하여 정적 분석 실행
- 한계
  - 페이지 요청 권한이 없거나 감춰져 있는 파일을 찾을 수 없을 때 발생하는 미탐

결론과 추후 연구

- 결론
  - 동적 테스팅으로 분석하고자 하는 페이지 요청
  - 동적으로 페이지를 분석하여 자바스크립트 추출
- 추후 연구
  - 취약점의 지속적인 발굴과 이에 대응한 분석 영역의 확장 필요


End&And NeverEndingStory © 2005 HAPPYSKY
end
and