웹 서비스 취약점 자동 진단 및 반자동 패치

AJOU SOFTCON 2021-1

팀 명 WeakEnd

팀 원 정지운, 이찬호, 황세정, 김강년

지도교수 손태식

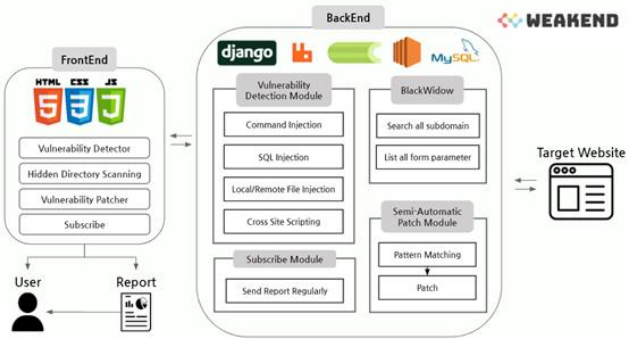
멘 토 시큐브 이규호

개발 동기 및 목적

최근 스타트업과 주니어 개발자의 수가 증가하면서 보안 문제에 관한 관심 또한 증가하고 있다. 스타트업의 경우 개발에 치중하는 경향이 크고, 보안 관련 예산 혹은 인력이 부족하여 체계적인 보안 시스템을 갖추기 어렵다. 스타트업의 이러한 보안 문제로 인해 해킹범들의 표적이 되어 개인정보가 유출되는 사건이 잇따라 발생하고 있는 상황이다. 그 중에서도 웹은 비교적 외부에서의 접근이 쉽기 때문에 웹 취약점에 따른 개인 정보 유출 사고가 보안 사고의 많은 비율을 차지하고 있다. 웹 취약점에 의한 개인 정보 유출 사고가 빈번히 발생하는 상황에서 보안 문제를 개인적 측면에서 해결할 수 있다면, 보안 사고를 감소시킬 수 있을 것이라는 생각에서 프로젝트를 진행하였다. 간편한 웹 사이트 보안 강화를 위해 웹사이트에 존재하는 취약점들을 자동으로 진단하여 보고서를 작성해주고, 더 나아가 발견한 취약점을 반자동으로 패치해주는 서비스를 개발하는 것을 목표로 한다.

취약점을 진단하기에 앞서 웹사이트 내에 존재하는 모든 서브 도메인, Dynamic URL, 페이지 내에 존재하는 URL들을 찾아 탐지 대상 URL을 확보한다. 이후 확보된 URL을 대상으로 다양한 우회 방법이 적용된 페이로드를 사용해 SQL Injection, XSS(Cross Site Scripting), File Inclusion, Command injection, Directory Traversal 공격을 Fuzzing 방식으로 수행한다. 수행된 공격 결과를 바탕으로 대상 웹 사이트의 취약점을 진단하고 취약점 보고서를 작성하여 사용자에게 제공한다. 더 나아가 탐지된 취약한 부분의 백엔드 코드를 사용자에게 입력 받아 시큐어 코딩이 적용된 코드와 언어 및 취약점별 상세한 패치 가이드라인을 출력해주는 반자동패치 기능을 제공한다. 또한 구독 기능을 통해 주기적으로 설정한 웹 사이트에 대해 자동 취약점 진단을 실시하여 취약점 진단 보고서를 메일로 전송해 준다.

개발 내용



BackEnd는 AWS EC2를 기반으로 Django 프레임워크를 사용해 동작한다. Blackwidow, 취약점 진단, 반자동 패치, 구독 기능 모듈들을 각각 Python으로 구현하였으며, 취약점 진단 도중 현재까지 발견한 취약점을 볼 수 있도록 Ajax를 사용했다. 백그라운드에서의 일처리를 Message Queue 형태로 구현하기 위해 RabbitMQ와 Django-Celery를 사용하였다. 이때 Celery의 Thread를 사용할 수 있는 기능을 통해 프로세스의 효율성을 증가시켰다. 각 모듈은 다음과 같이 동작한다. BlackWidow 모듈은 Target URL에서 서브 도메인, 디렉토리 리스트를 분석해 URL List를 만든다. 취약점 진단 모듈은 URL List를 대상으로 Request를 통해 각 취약점에 대한 퍼징 기반 공격을 수행한다. 반자동 패치 모듈은 취약 코드 패턴을 파악하여 안전한 코드로 수정한다. 구독 모듈은 Crontab 기능을 통해 특정 기간마다 사용자에게 진단 보고서를 발송한다.

주요 기술

- **BlackWidow**
  beautiful soup을 통해 서브도메인, dynamic URL, form 태그가 존재하는 URL들을 분류하여 txt파일로 저장한다.
- **취약점 탐지**
  각 취약점의 공격 페이로드를 전송한뒤 획득한 응답에서 다음 사항을 확인하여 취약점 존재여부를 진단한다.
  SQL Injection
  Blind based, Error based, Union based 공격 성공 여부 탐지를 통해 취약점 존재 여부를 진단한다.
  - Local/Remote File Inclusion
    특정 파일 내용의 패턴, 특정 해쉬값 존재 여부 확인을 통해 취약점 존재 여부를 진단한다.
  - Cross-site Scripting (XSS)
    이스케이핑 되지 않은채 사용한 페이로드가 그대로 발견될 경우, 알람을 보내는 스크립트 패턴이 존재할 경우를 기준으로 취약점 존재 여부를 진단한다.
  - Command Injection
    ls, netstat, cat /etc/passwd 등의 명령어 결과 패턴을 검색하여 취약점 존재 여부를 진단한다.
- **반자동 패치**
  패치 전 코드에 취약 코드 패턴이 존재하는지 확인 후 존재할 경우, 패치에 사용할 함수를 삽입한 뒤 패치 패턴에 맞추어 패치를 진행한다.
- **구독**
  Crontab 기능으로 쉘 스크립트를 사용하여 일정 기간마다 딕셔너리 형태로 변환한 Target URL을 진단 후 구독을 신청한 사용자의 메일로 취약점 진단 보고서를 전송한다.

결과 및 분석

- **결과**
  - **DVWA**
    DVWA 내 SQL Injection, Local/Remote File Inclusion, XSS,Command Injection 취약점 탐지 가능
  - **아주대학교 대상**
    진단 범위: museum.ajou / ajou.ac.kr / security.ajou
    일시: 2021.05.26~2021.05.30 (오후 11시 ~ 오전 3시)

    발견 취약점 리스트

    ▪ **SQL Injection**
    database 조작 및 삭제, DB 내부의 데이터를 가져올 수 있는 취약점 탐지
    ➢ Time-based Injection, Boolean-based Injection

    ▪ **Directory Traversal**
    일반 유저는 탐색이 제한되어야 할 페이지 확인
    ➢ 아주대 사이트 내 소스파일들을 가져오거나 원격 코드 실행 공격까지 실행 가능한 취약점 발견

    ▪ **Security Misconfiguration**
    잘못된 보안 설정으로 서버 내부 정보 유출

- **발전 방향**
  - 탐지 가능한 취약점 종류의 증가
  - 탐지 알고리즘 개선
  - 반자동 패치 알고리즘의 머신러닝/딥러닝 활용