웹 퍼징을 이용한 취약점 분석

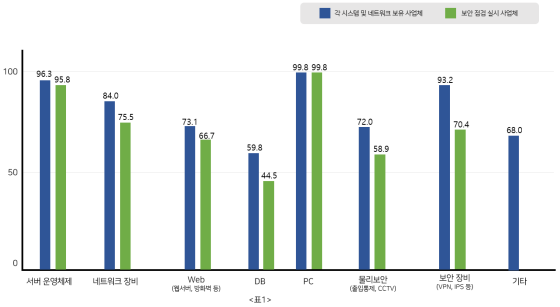
강민영, 이경서, 이유경, 장혜선

요 약

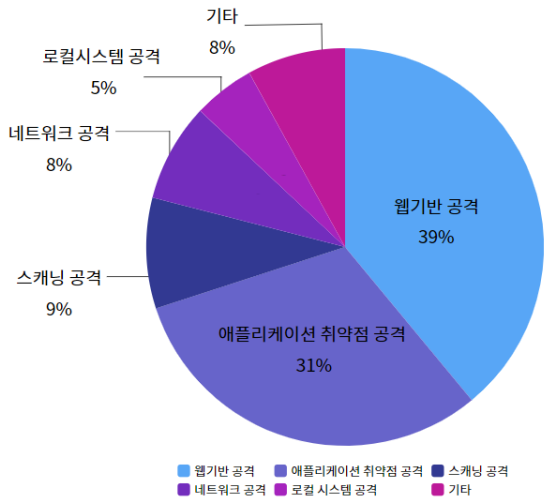
정보통신 인프라가 급속하게 확장되면서 대부분의 기업, 기관 등은 자신만의 웹 사이트를 운영하고 있다. 사용자가 증가함에 따라 웹 사이트에 악의적인 웹 공격으로 접속자들의 정보가 노출되는 등의 정보 안정성에 대한 위협이 높아지고 있다. 본 논문에서는 주요 웹 취약점 유형, 취약점에 대한 진단 기준 및 조치 방법 등에 대해 알아보고, 보안 방안이 적용된 사이트에서 다양한 웹퍼징을 통해 취약점을 수집한 뒤 미조치율이 높은 상위 세 항목에 대한 보안 방안을 선제적으로 제시한다.

Ⅰ.서론

정보통신 인프라가 급속하게 확장되면서 대부분의 기업, 기관 등은 자신만의 웹 사이트를 운영하고 있으며, 대다수의 사용자들은 웹 사이트에 접속함으로써 다양한 정보를 얻는다. 그러나 웹 사이트에 악의적인 웹 공격으로 접속자들의 정보가 노출되는 등의 정보 안정성이 위협을 직면할 수 있다. 웹 사이트의 발달로 정보보안 3요소의 중요성이 대두 되며, 시중에서 제공되는 보안 가이드 혹은 대응책을 적용하고 있다.



<표1>



<표1> 과학기술정보통신부 통계치에 따르면 보안점검 실시 사업체 중 Web 관련 취약점을 점검하는 사업체는 66%이고, Web 시스템을 보유하고 있는 사업체는 73%에 이른다. 많은 업체에서 보안 점검을 실시했음에도 불구하고 <표2> 안랩에서 발표한 2019년 사이버 공격 통계에서 웹 기반 공격이 39%로 가장 큰 비율을 차지했다.

본 논문에서는 보안 방안을 적용한 사이트에서 웹 취약점 점검을 통해서 취약점을 찾아 분석한다. 대상은 국내 대학 홈페이지로 선정하였으며, 웹 퍼징을 이용하여 취약점을 수집하고, 수집된 취약점의 공통점으로 그룹화하여 분석 및 원인을 파악하고자 한다.

세부항목 별 미조치율

세부 항목	미조치율
대체 경로를 통한 인증 우회	50%
취약한 패스워드 복구	50%
하드코딩된 암호화 키 사용	36%
불필요한 서비스	29%
계정정보, 개인정보 등 민감 데이터 평문 전송	29%
에러메시지를 통한 정보 노출	23%
관리자 페이지 노출	22%
악의적인 스크립트 실행	22%
취약한 쿠키 관리	22%
URL/파라미터 변조	21%
웹 서비스 메소드 설정 공격	21%
취약한 패스워드 사용	17%
다운로드 기능을 악용한 비정상 접근	16%
백업 및 예제 파일	15%
실명인증 우회	14%
악성파일 업로드	14%
인젝션 실행	5%
디렉터리 인덱싱	0%
취약한 세션 관리	0%
쿠키/세션 값 변조를 통한 인증/권한 우회	0%

<표3>

<표3>은 웹 공격 항목별 미조치율을 나타낸다. 본 논문에서는 <표3>의 상위 세 항목에 대한 보안 방안을 선제적으로 제시한다.

본 논문에서는 주요 웹 취약점 유형, 취약점에 대한 진단 기준 및 조치 방법 등에 대해 알아보고, 보안 방안이 적용된 사이트에서 다양한 웹퍼징을 통해 취약점을 수집한다. 또한, 해당 취약점들의 공통점을 분석하여 원인을 파악하여 최종적으로 대처 방안까지 모색한다.

Ⅱ. 관련 연구

2.1 퍼징

퍼징이란 소프트웨어의 취약점 테스트 방안 중 하나로, 유효하지 않은 값이나 임의의 값을 프로그램에 입력함으로써 예기치 못한 오류나 충돌을 일으킨다. 발생한 오류를 분석하는 과정에서 소프트웨어나 시스템에 존재하는 코드 상의 결함을 찾아낸다.

2.2 퍼징의 필요성

퍼징의 필요성은 크게 개발 측면과 보안 측면으로 나눌 수 있다.

개발 측면은 퍼징 기법의 적용으로 개발 중인 웹사이트의 취약점을 미리 찾아 예방 할 수있다. 그래서 제품의 신뢰도를 잃지 않게끔 해준다. 또한, 소프트웨어에 산정되는 전체 비용을 대폭 감소시켜주는 효과를 가진다. 초기 단계에서부터 적용한다면 버그 처리 비용을 극소화 시킬 수 있다.

보안 측면은 퍼징 기법의 적용으로 개발 단계에서 예측하지 못한 결함을 사전에 예방하는 데 있어서 큰 도움을 준다. 비정상적인 입력값이나 예상치 못한 위협을 예방하는 수단으로 미래에 발생할 치명적인 보안 위협을 예방 할 수있다.

2.3 퍼징 기법

퍼징 기법은 입력 값을 조절하는 방법에 따라 크게 두 가지 방식으로 나뉜다. Mutation(변이) 기법은 새로운 테스트 케이스를 생성하기 위해서 많은 입력 값을 넣어보며, 새로운 입력 데이터를 생성한다. 넣어지는 입력 값을 씨드 데이터라고 통칭한다. 데이터 형식이나 구조에 대한 명확한 이해가 어려울 때 사용하고, 씨드 데이터 무작위로 생성이 되어 필요하지 않은 입력 데이터가 생성되는 경우가 많기 때문에 Dumb(멍청이) Fuzzing이라고 불리기도 한다.

또 다른 기법으로는 Generation(생성 기반) 기법이 있다. 다른 말로는 Smart Fuzzing이라고 불린다. Mutation 기법과는 다르게 데이터 형식이나 구조 및 프로토콜을 이해하여 적절한 입력 값을 생성할 수 있다. 모델을 분석하여 포맷에 맞춰서 구현을 해야하므로 난이도가 높고 많은 시간이 소요되기도 한다. 그러나 불필요한 입력 데이터 생성을 최소화할 수 있다는 점에서 효율성을 가진다.

2.4 한계점

퍼징이 매우 효과적인 버그 탐지법임에도 불구하고 한계가 존재한다. 특히, Black-Box Testing나 간단한 버그를 찾을 때는 유용하나, 내부적으로 문제점이 있다. 랜점 접근 방식에서의 랜덤이라는 것이 장점이지만 경계 값에서 발생하는 이슈를 찾기 힘들며, 퍼징의 영향을 확인할 정보가 다양하지 않다는 점에서 본질적인 한계를 갖는다.

Ⅲ. 제안

3.1 기법

3.1.1 대체 경로를 통한 인증 우회

대체 경로를 통한 인증 우회 기법은 인증 절차가 불충분할 경우, 민감 데이터 접근이 가능한 경로에서 발생하는 취약점으로, 약 0.5%의 상대적으로 낮은 발견 비율임에도 불구하고 미조치율이 50%로 가장 높은 기법이다.

3.1.2. 취약한 패스워드 복구

취약한 패스워드 복구 기법은 불충분한 인증 및 인가로 인해 발생하는 취약점으로, 비밀번호 찾기 등과 같은 복구 로직으로 인하여 공격자가 피해자의 패스워드를 획득 및 변경할 수 있는 취약점이다. 이는 50%의 미조치율로 상대적으로 취약점 조치가 힘든 항목이다.

3.1.3. 하드 코딩된 암호화 키 사용

하드 코딩된 암호화 키 사용 기법은 공격자에게 2차 공격을 하기 위한 데이터가 제공되도록 정보가 암호화 되거나 함수로 처리되지 않고 소스에 코딩되는 취약점이다. 이는 <표3>에서 나타난 취약점 중 미조치율 36%로 세번째로 높은 비율을 차지한다.

3.2 제안

<표3>을 중점으로 미조치율이 높은 공격 3가지 공격에 대한 공통점을 분석해 취약점에 대한 조치 방안 대응 모델을 제안한다.

Ⅴ. 결론 및 향후 연구

4.1 조치 방안

4.2 대응 모델링 적용

Ⅵ. 참고문헌

[1] 김연재, 최종학, 신명재 이인회, 이재호, 최자운, 박지환, “취약점 분석을 위한

퍼징(Fuzzing)”, 2010.10, http://hisjournal.net/doc/[KUCIS_Project]_Fuzzing_for_Finding_Vulnerabilities_by_CERT-IS.pdf
[2] 성종혁, 이후기, 고인제 and 김귀남. (2018). 웹 취약점 점검 및 취약점별 조치 우선순위 산정에 관한 연구. 융합보안 논문지, 18(3), 37-44.
[3] 김평화 "2019년 웹 취약점·미디어 대상 사이버 공격 시도 두드러졌다" "2019년 웹 취약점·미디어 대상 사이버 공격 시도 두드러졌다" - IT조선 > 기술 > 보안 (chosun.com)
[4] 통계청 “각 시스템 및 네트워크 보유 사업체 취약점 점검 https://kosis.kr/statHtml/statHtml.do?orgId=127&tblId=DT_342005_A064&conn_path=I2
[5] 박영웅(Yeongung Park),이준혁(Junhyuk Lee),and 조성제(Seongje Cho). "퍼징 도구들의 비교 분석을 통한 효율적인 웹 브라우저 퍼징 전략." 한국정보과학회 학술발표논문집 38.1A (2011): 328-331.
[6] 전소희, 이영한, 김현준, 배윤홍 “최근 퍼징 기법들과 발전에 관한 연구 『2020 온라인 춘계학술발표대회 논문집 제27권 제1호 (2020. 5)』
[7]서진원,서희석,곽진,Seo Jin-Won,Seo Hee-Suk,and Kwak Jin. "웹서비스 공격정보 분류 방법 연구." 한국시뮬레이션학회 논문지 19.3 (2010): 99-108.