7주차 1차시 웹해킹

【학습목표】

1. 웹해킹의 개요와 웹스캔의 개념을 설명할 수 있다.
2. 웹프록시와 취약점을 분석할 수 있다.

학습내용1 : 웹해킹의 개요와 웹스캔

1. 웹 해킹

<웹 사이트의 구조와 동작 원리를 이해하는 것에서부터 시작>
- 실제로 웹의 모의해킹 과정에서 초기의 몇 일간은 해당 사이트를 만든 사람의 코딩 스타일, 사이트 구조, 습관, 인수 전달 방식 등을 파악함

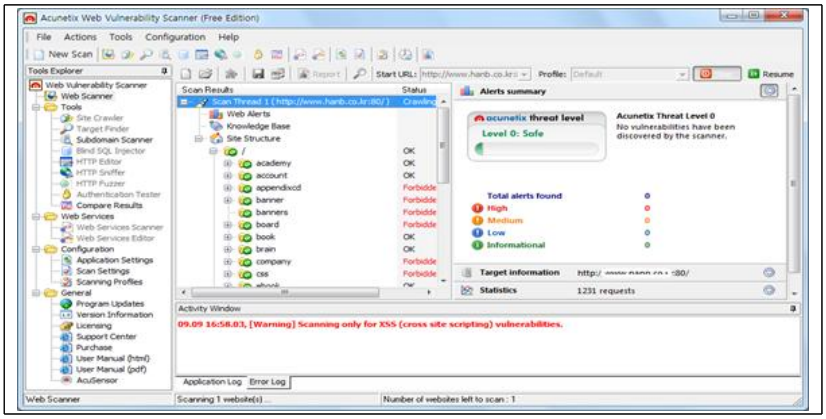
2. 웹 스캔, 웹 프록시를 이용한 패킷 분석 구글 해킹 등

- 웹 해킹에 소요되는 대부분의 시간을 차지할 만큼 중요한 과정임
- 웹 사이트에 대한 이해만 제대로 수행하면 웹 해킹은 무척 쉬움

3. 웹 취약점 스캐너를 통한 정보 수집

① 장점 : 웹 취약점 스캐너를 통한 정보 수집은 빠른 시간 내에 다양한 접속 시도를 수행할 수 있음
② 단점 : 웹 구조를 파악하고 취약점을 수집하기가 쉽지 않음

1) [그림] Acunetix 웹 취약점 스캐너



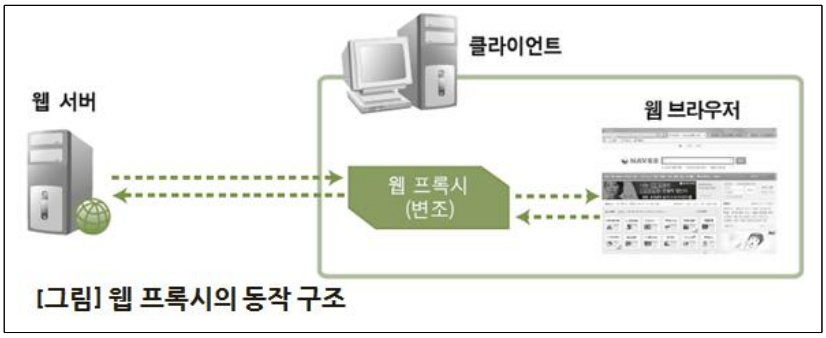
- 각 페이지에서 링크 정보를 따라가며 웹 사이트의 웹 페이지를 로컬에 저장해주는 Web Zip이라는 프로그램
- 웹 스캔은 Web Zip 이 웹 페이지를 수집하는 원리와 같음
- 웹 취약점 스캐너를 통해 확인된 취약점이 실제로 취약점이 존재하는 경우도 있지만 그렇지 않은 경우도 많음

- 웹 프로그램은 자유스럽게 만들어지고 변형도 다양해 웹 스캐너가 취약점을 정확히 잡아내기란 거의 불가능

학습내용2 : 웹 프록시와 취약점분석

1. 웹 프록시를 통한 취약점 분석

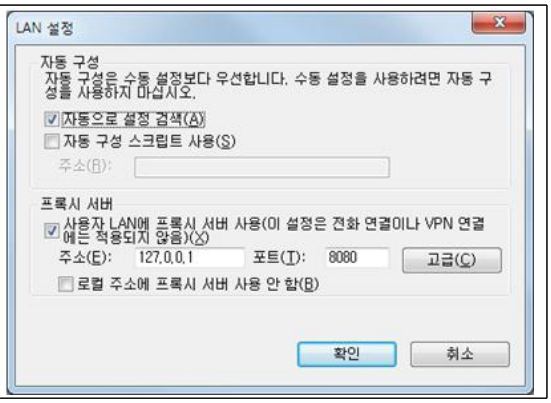
<웹 프록시는 클라이언트가 웹 서버와 웹 브라우저 간에 전달되는 모든 HTTP 패킷을 웹 프록시를 통해서 확인하면서 수정하는 것이 가능>



[그림] 웹 프록시의 동작 구조

1) 웹 프록시로 burp suite를 사용해보자
- [도구]-[인터넷 옵션]-[연결]-[LAN 설정]에서 프록시 서버를 다음과 같이 설정해주어야 함
- 127.0.0.1을 흔히 루프백(Loopback) 주소라 하는데, PC 자기 자신을 의미함
- 8080은 웹 프록시 프로그램의 서비스 포트

[그림]
웹 프록시의 설정
: [LAN 설정]
-[프록시 서버]



2) 서버에서 클라이언트로 전송되는 패킷 변조
* 웹 사이트가 언어로 개발됐든지 웹 프록시를 통해서 확인하는 것은HTML
* 테스트환경으로 사용하는 웹 페이지의 게시판에서 하나의 글에 대한 열람을 시도해보자

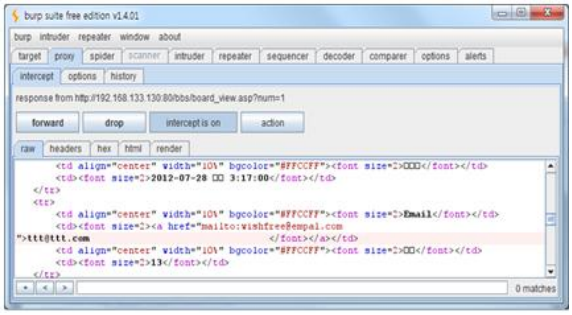
[그림]
'테스트1입니다.'
문서 열람 시도

* '테스트1입니다.'라는 제목을 클릭하면 다음과 같은 내용을 확인할 수 있음

[그림] 열람한 '테스트1입니다.'의 내용

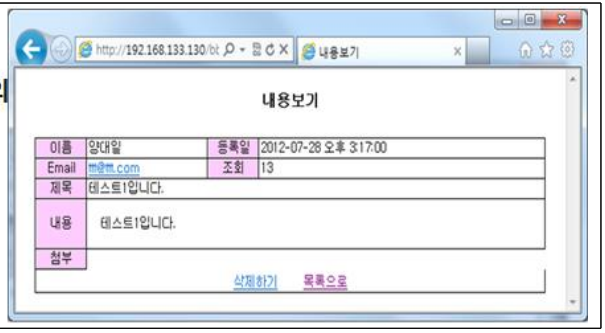
* Email 값을 tt@ttt.com으로 바꿔보자

[그림] 웹 프록시를 통해 Email 변조



* 전송해보자

[그림] Email을 변조한 '테스트1입니다.'의 내용

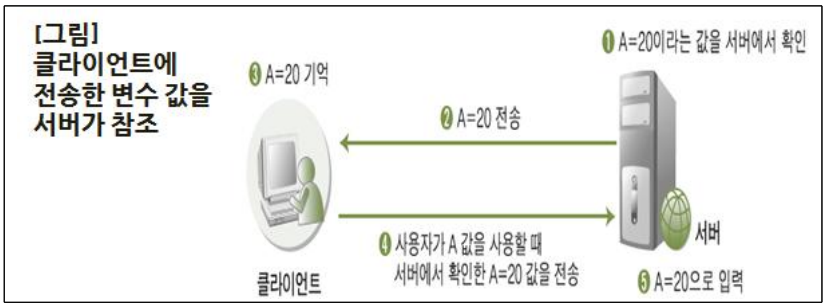


3) 서버에서 클라이언트로 전송되는 패킷 변조

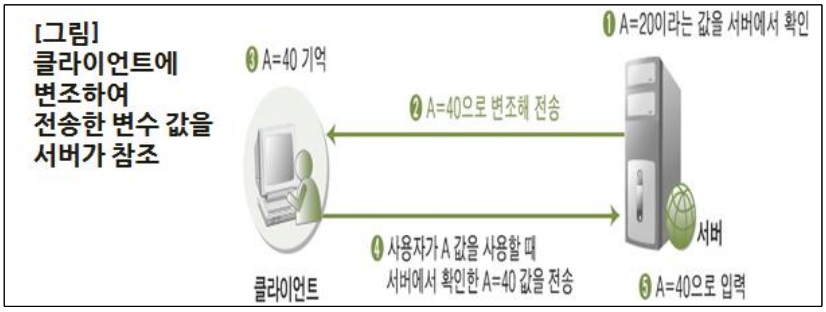
* 클라이언트에 해킹하고자 하는 대상이 있는 경우

- 웹 브라우저 내용만 바꾸었지만 실제로는 Active X 등의 형태로 여러 프로그램이 클라이언트에 설치되어 웹 서비스를 제공하는 경우가 많음
- 이때 클라이언트에 설치된 서비스 프로그램을 속이는 것이 가능

* 서버에서 클라이언트에 정보를 전송했다가 이를 다시 전송 받아 처리하는 경우
- 예시 : 서버에서 변수 A의 값이 20임을 확인하고 이 값을 클라이언트에 전송
- 그리고 서버는 전송한 변수 A가 필요할 때 자신의 데이터베이스에서 다시 읽지 않고, 클라이언트가 관련 서비스 수행할 때 서버에 다시 전송해주는 A 값을 참조하여 서비스를 수행하는 경우

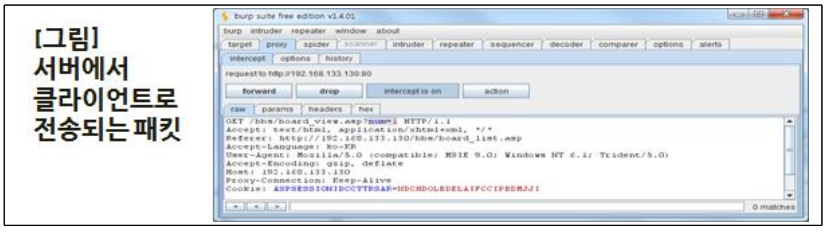


* 2단계에서 A=40이라고 바꾸어 전송하면 A 값이 다음과 같이 흘러감



예시) 패킷 변조의 예

4) 클라이언트에서 서버로 전송되는 패킷 변조
<'테스트1입니다.'의 글을 조회하는 과정에서 HTTP 패킷을 웹 프록시에서 확인해보자>
- 해당 글에 대한 인수값(num=1)이 전달되는 것을 확인할 수 있음

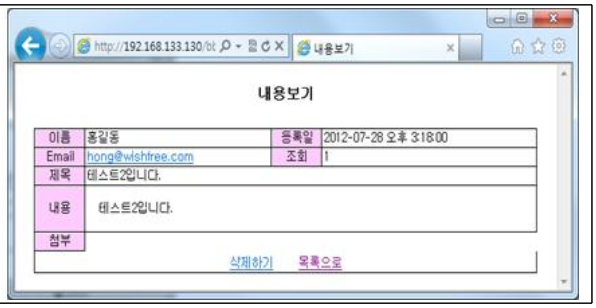


<GET을 통해서 게시판의 첫 번째 글 /bbs/board_view.asp?num=1을 보여줄 것을 서버에 요청>
- num 값을 2로 바꾸어 전송

[그림]
서버에서
클라이언트로
전송되는
패킷 변조

<패킷을 보내면 2번 글이 다음과 같이 조회되는 것을 확인할 수 있음>

[그림]
변경된 본문 내용



- 클라이언트에서 서버로 전송되는 패킷을 변조하는 것은 일반적인 웹 서비스의 메뉴상 접속할 수 없는 것에 접근하거나 특정한 값을 넣어 시스템의 오작동을 유도하기 위한 목적으로 사용

5) 구글 해킹을 통한 정보 수집
<많은 정보를 수집하기 위해서는 검색 엔진을 이용하면 유용>
- 검색 엔진 중에는 구글이 많이 사용됨

[표] 구글에서 제공하는 고급 검색 기능

검색 인자	설명	검색 추가 인자
site	특정 도메인으로 지정한 사이트에서 검색하려는 문자열이 포함된 사이트를 찾음	YES
filetype	특정한 파일 유형에 한해서 검색하는 문자가 들어 있는 사이트를 찾음	YES
link	링크로 검색하는 문자가 들어 있는 사이트를 찾음	NO

[표] 구글에서 제공하는 고급 검색 기능		
검색 인자	설명	검색 추가 인자
cache	특정 검색어에 해당하는 캐시된 페이지를 보여줌	NO
intitle	페이지의 제목에 검색하는 문자가 들어 있는 사이트를 찾음	NO
inurl	페이지의 URL에 검색하는 문자가 들어 있는 사이트를 찾음	NO

* 주요 검색 인자

① site

- 특정 사이트만을 집중적으로 선정해서 검색할 때 유용
- wishfree.com 도메인이 있는 페이지에서 admin 문자열을 찾으라는 예

```
site:wishfree.com admin
```

② filetype

- 특정 파일 유형에 대해 검색할 때 사용함
- 파일 확장자가 txt이고 패스워드라는 문자열이 들어간 파일을 검색한 화면

```
filetype:txt 패스워드
```

[그림] filetype 기능의 예제 결과

③ intitle

- 디렉터리 리스팅 취약점이 존재하는 사이트를 쉽게 찾을 수 있음

```
intitle:index.of admin
```

[그림] 디렉터리 리스팅이 가능한 사이트 검색

* 검색 엔진의 검색을 피하는 방법
- 가장 일반적인 대응법
    - 웹 서버의 홈 디렉터리에 robots.txt 파일을 만들어 검색할 수 없게 만듦
    - http://www.wishfree.com/robots.txt 파일이 있으면 구글 검색 엔진은 robots.txt에 있는 디렉토리는 검색하지 않음
    - robots.txt 파일은 User-agent와 Disallow를 이용

* User-agent는 구글 검색 엔진으로부터의 검색을 막기 위해서 다음과 같이 사용

User-agent: googlebot
→ 구글 검색 엔진의 검색을 막음

User-agent: *
→ 모든 검색 로봇의 검색을 막음

Disallow: dbconn.ini
→ dbconn.ini 파일을 검색하지 못하게 함

Disallow: /admin/
→ admin 디렉터리에 접근하지 못하게 함

* 미국 백악관에서 실제로 사용하는 robot.txt 파일의 내용을 살펴보는 것도 좋음
- http://www.whitehouse.gov/robots.txt

[그림]
www.whitehouse.gov/robots.txt의 내용

【학습정리】

1. 웹 취약점 스캐너는 웹 취약점 스캐너를 통한 정보 수집은 빠른 시간 내에 다양한 접속 시도를 수행할 수 있는 장점과 웹 구조를 파악하고 취약점을 수집하기가 쉽지 않은 단점이 있다.

2. 웹 프록시를 통한 취약점 분석은 클라이언트가 웹 서버와 웹 브라우저 간에 전달되는 모든 HTTP 패킷을 웹 프록시를 통해서 확인하면서 수정하는 것이 가능하다.

3. 많은 정보를 수집하기 위해서는 검색 엔진을 이용하면 유용한데 구글은 인자를 통해 확장검색을 이용하므로 정보수집에 사용된다.