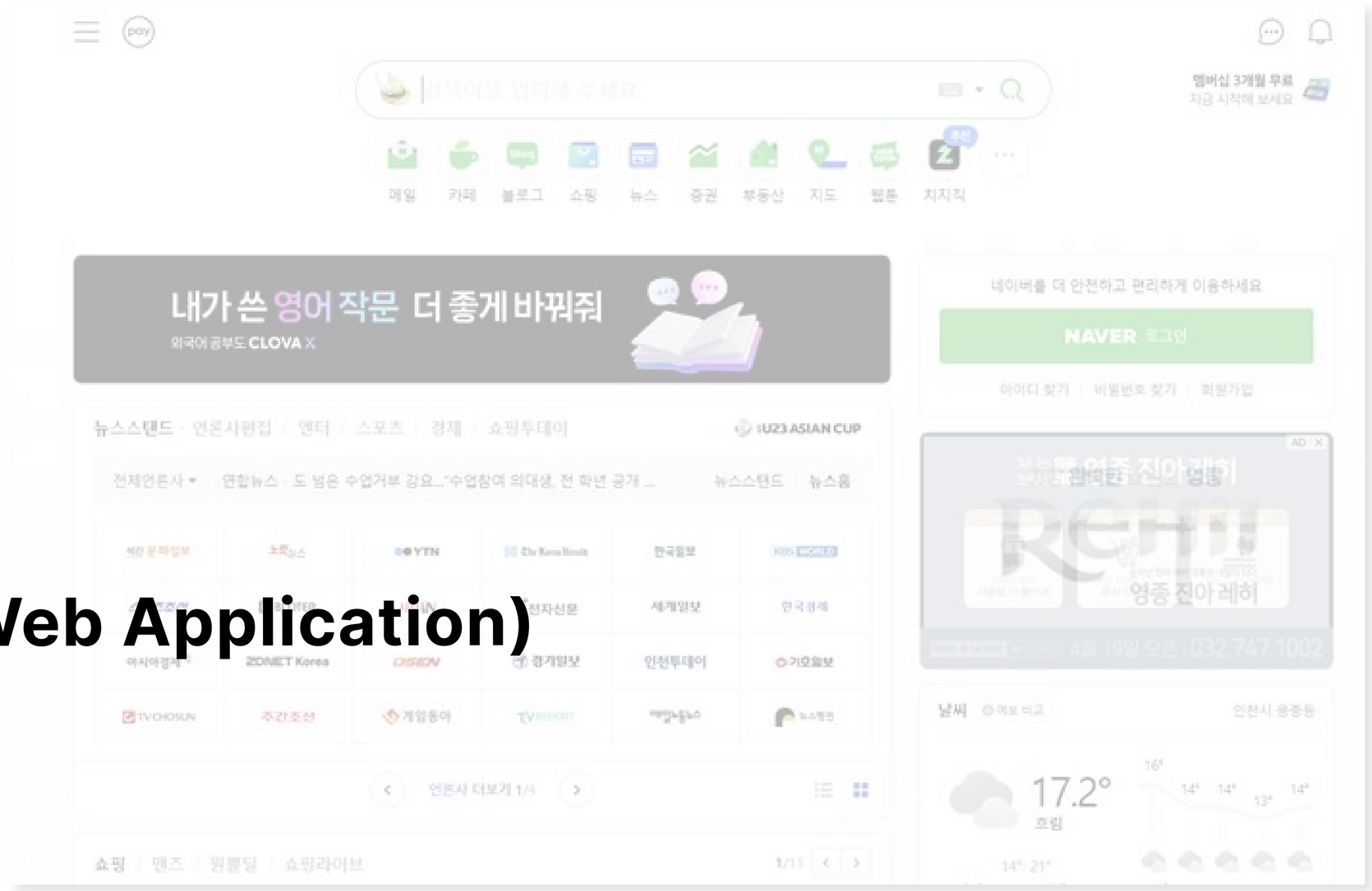
웹 취약점 (1)

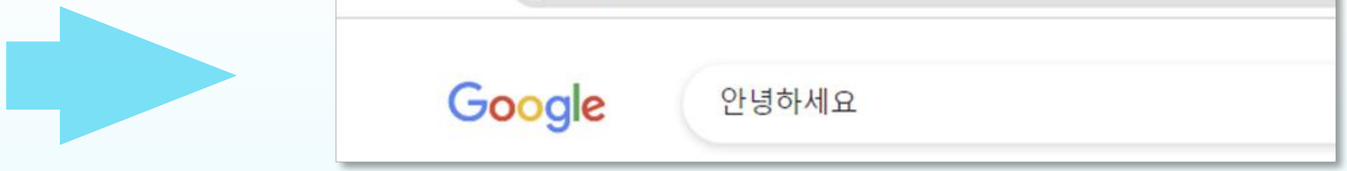
웹 애플리케이션(Web Application)

- 웹을 이용한 애플리케이션

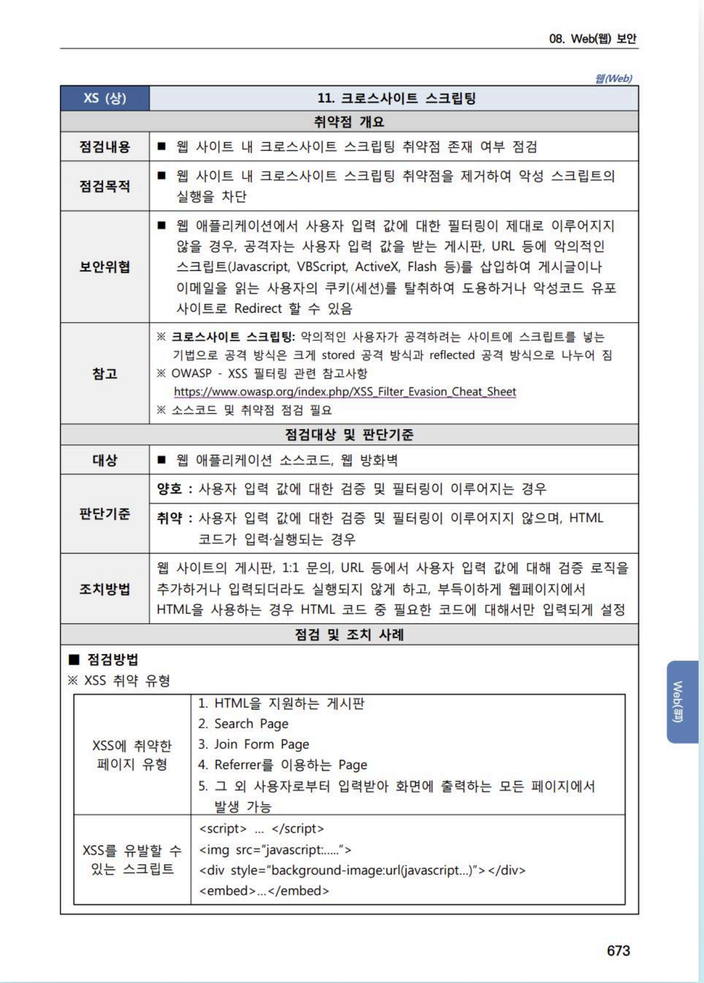


웹의 주요 공격포인트

- 개인정보 조회/수정 페이지
- 금융/포인트/결제/증명서 등 핵심 서비스
- 게시판, 문의사항, FAQ 등
- 관리자 페이지
- 입력 값이 존재하는 모든 URL



웹 취약점 분석 평가(Vulnerability Assessment)



08. Web(웹) 보안

웹(Web)

XS (상)	11. 크로스사이트 스크립팅
취약점 개요	
점검내용	■ 웹 사이트 내 크로스사이트 스크립팅 취약점 존재 여부 점검
점검목적	■ 웹 사이트 내 크로스사이트 스크립팅 취약점을 제거하여 악성 스크립트의 실행을 차단
보안위협	■ 웹 애플리케이션에서 사용자 입력 값에 대한 필터링이 제대로 이루어지지 않을 경우, 공격자는 사용자 입력 값을 받는 게시판, URL 등에 악의적인 스크립트(Javascript, VBScript, ActiveX, Flash 등)를 삽입하여 게시글이나 이메일을 읽는 사용자의 쿠키(세션)를 탈취하여 도용하거나 악성코드 유포 사이트로 Redirect 할 수 있음
참고	※ **크로스사이트 스크립팅:** 악의적인 사용자가 공격하려는 사이트에 스크립트를 넣는 기법으로 공격 방식은 크게 stored 공격 방식과 reflected 공격 방식으로 나누어 짐 ※ OWASP - XSS 필터링 관련 참고사항 https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet ※ 소스코드 및 취약점 점검 필요
점검대상 및 판단기준	
대상	■ 웹 애플리케이션 소스코드, 웹 방화벽
판단기준	**양호 :** 사용자 입력 값에 대한 검증 및 필터링이 이루어지는 경우
	취약 : 사용자 입력 값에 대한 검증 및 필터링이 이루어지지 않으며, HTML 코드가 입력·실행되는 경우
조치방법	웹 사이트의 게시판, 1:1 문의, URL 등에서 사용자 입력 값에 대해 검증 로직을 추가하거나 입력되더라도 실행되지 않게 하고, 부득이하게 웹페이지에서 HTML을 사용하는 경우 HTML 코드 중 필요한 코드에 대해서만 입력되게 설정
점검 및 조치 사례	

■ **점검방법**

※ XSS 취약 유형

XSS에 취약한 페이지 유형	1. HTML을 지원하는 게시판 2. Search Page 3. Join Form Page 4. Referrer를 이용하는 Page 5. 그 외 사용자로부터 입력받아 화면에 출력하는 모든 페이지에서 발생 가능
XSS를 유발할 수 있는 스크립트	<script> ... </script> <img src="javascript:....."> <embed>...</embed>

673

Web(웹)

- 체크리스트 기반으로 웹사이트를 점검
- 발견된 취약점에 대한 평가(취약항목 및 위험도 산정)
- 발견된 취약점에 대한 조치 가이드
- 다양한 취약점을 식별하고 평가
- 최대한 많은 취약점을 식별
- 법(컴플라이언스, Compliance) 관점

기본 점검 항목

- 전자금융기반시설 취약점 분석 평가 항목 – 금융회사 사용
- 주요정보통신기반시설 취약점 분석평가 항목 - 공공기관/공기업 등 사용

주요 항목들

- SQL Injection(SQL 인젝션)
- XXS(Cross Site Scripting, 크로스 사이트 스크립팅)
- CSRF(Cross-site request forgery, 사이트 간 요청 위조)
- 불충분한 인증/인가
- 정보노출, 명령실행(BoF, FSB), 비밀번호 관리, 세션/쿠키 관리
- 세션만료/고정, 파일업로드/다운로드, 관리자 페이지

08. Web(웹)

XSS(Cross Site Scripting)란?

What is ‘XSS’?

- Cross Site Scripting의 줄임말(줄여서 크사라고 부르기도 함)
- Cross-site(크로스 사이트) : 여러 사이트 간의 관계
- scripting(스크립팅): *스크립트

스크립트란?

- 컴퓨터에서 실행되는 명령어의 모음.
- 예) ‘로그인‘ 버튼을 누르면 화면에 입력한 정보를 서버로 보내는 ‘스크립트’가 실행됨

XSS의 원리

- 해커가 악의적인 스크립트(해킹 코드)를 웹사이트나 앱에 입력해서 공격하는 기법
- 보통 사용자들이 입력하는 데이터(검색어, 댓글, 이메일 등)을 이용해 이뤄질 수 있음
- 사용자 입력값(Request)을 그대로 출력(Response)해줄 때 발생
- 브라우저에 임의의 스크립트 코드 실행 – 대표적인 Client Side 공격
- 발견률이 매우 높은 취약점
- 다른 웹사이트와 정보를 교환하는 식으로 작동하기 때문에 사이트 간 스크립팅이라고도 부름

Java Script 코드 짚고 넘어가기

코드	설명
<script>	Javascript(자바스크립트) 코드의 시작
alert(“블라블라”)	웹 사이트에 메시지 창을 띄워 줌
</script>	Javascript(자바스크립트) 코드의 끝
;	코드 한 줄의 끝

예시)

<script> alert (“안녕하세요”) ; </script>

다른페이지로 이동하는 자바스크립트 코드

다른 웹페이지로 이동하기

예시)

네이버 웹사이트

XSS 실습

XSS가 먹히지 않는 안전한 웹사이트를 만들려면?

1. 입력 값 제한
   - http://www.qubitsec.io?page=1
   - 위와 같은 URL 에서 사용자의 입력 값인 page 가 화면에 바로 뿌려짐
   - 이때 page 의 값을 숫자만으로 허용한다면 그 외의 입력 값에는 출력되지 않기 때문에 대응

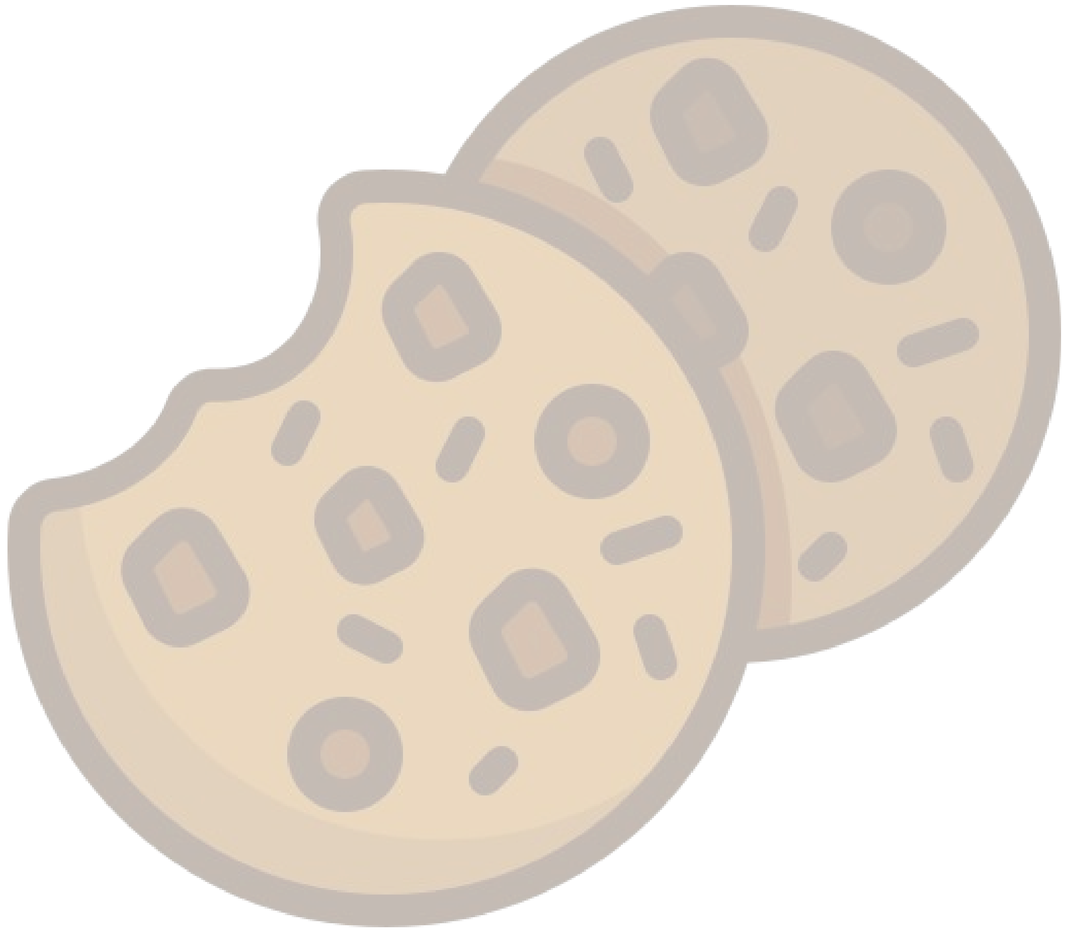
2. 입력 값 치환

3. 스크립트 영역에 출력 자제

ASCII 문자	참조 문자	ASCI 문자	참조 문자
&	&amp;	"	&quot;
<	&lt;	'	&#x27;
>	&gt;	/	&#x2F;
(	&#40;	)	&#41;

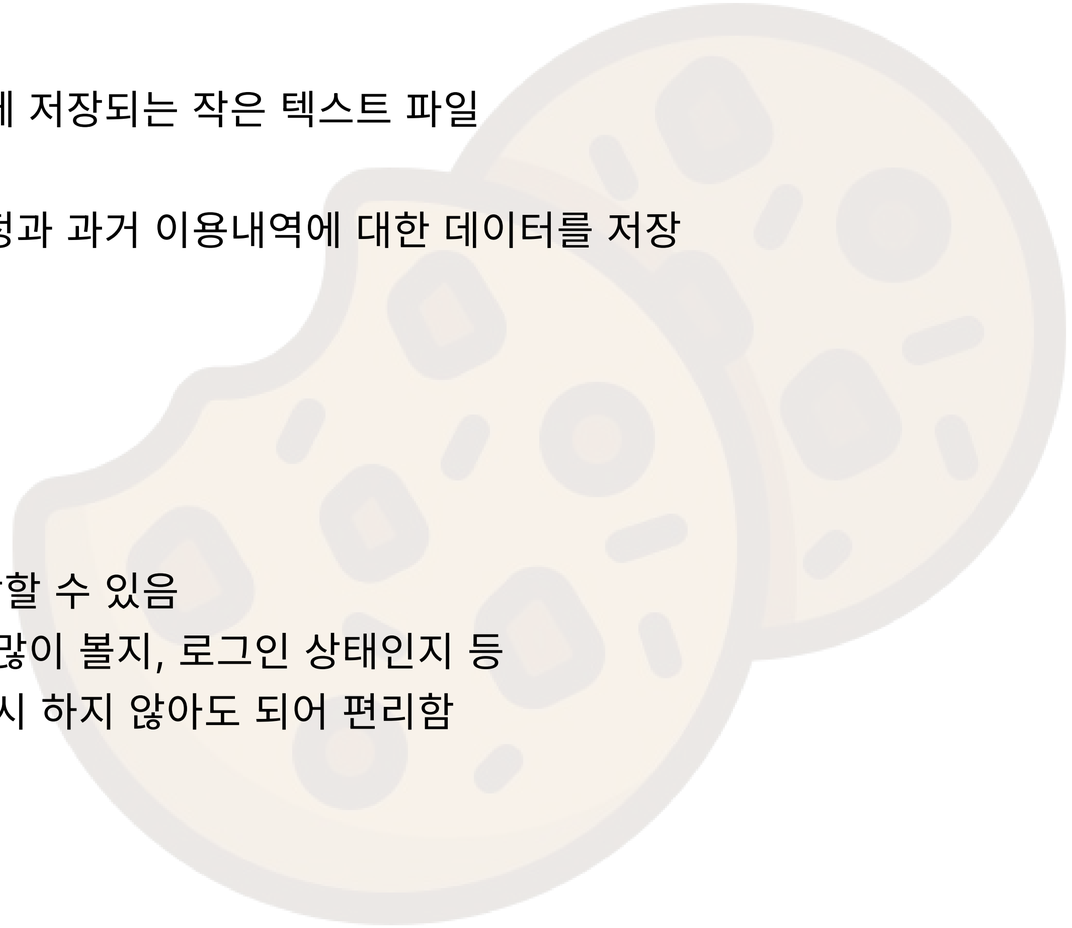
[그림 13] 위험 문자 인코딩

세션/쿠키 변조란?



What is ‘Cookie(쿠키)’?

- 웹사이트 접속 시 접속자의 개인장치에 다운로드 되고, 브라우저에 저장되는 작은 텍스트 파일
- 웹사이트와 컴퓨터 간의 대화를 도와주는 작은 메모지와 같음
- 웹사이트는 쿠키를 통해 접속자의 장치를 인식하고, 접속자의 설정과 과거 이용내역에 대한 데이터를 저장



쿠키를 사용하는 이유는?

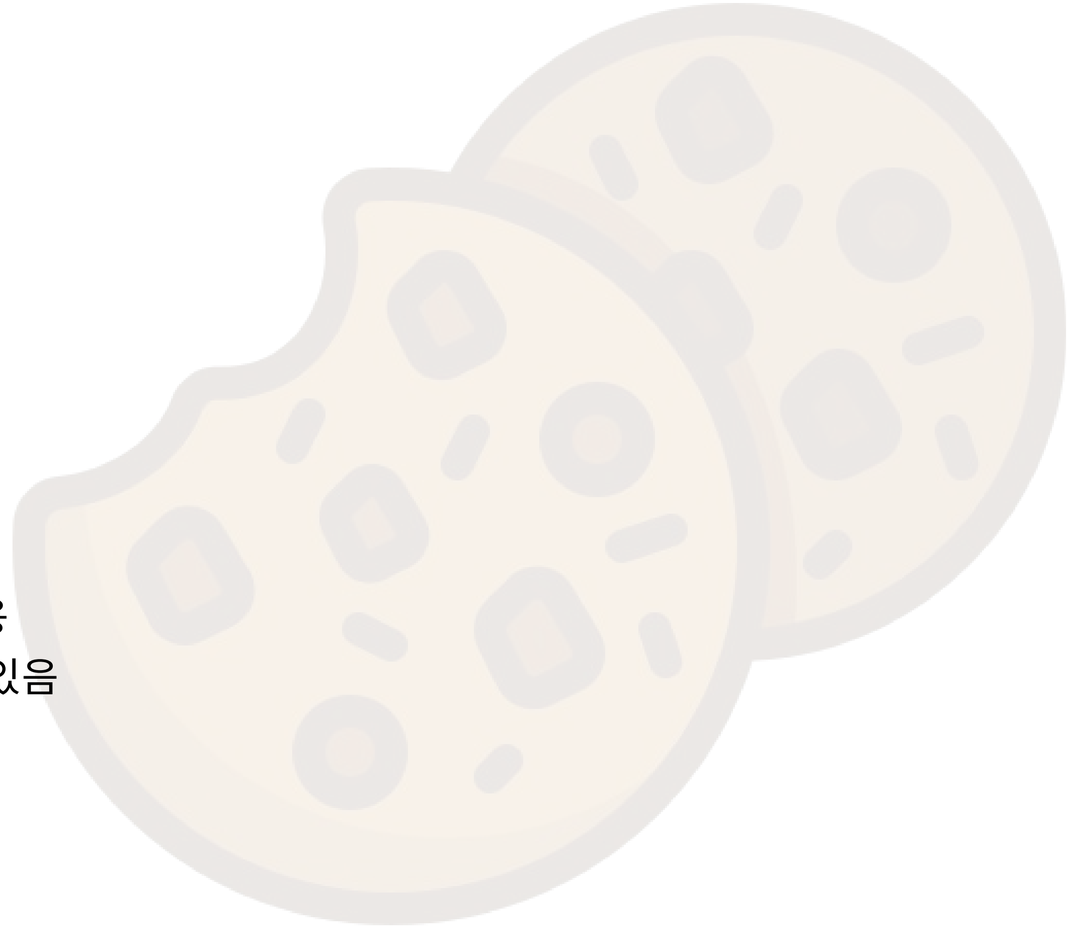
- 웹사이트는 쿠키를 사용해서 사용자에 대한 여러가지 정보를 저장할 수 있음
- 사용자가 사이트를 얼마나 자주 방문하는지, 어떤 페이지를 가장 많이 볼지, 로그인 상태인지 등
- 사용자는 다음에 같은 웹사이트를 방문할 때 이전에 한 설정을 다시 하지 않아도 되어 편리함
- 로그인을 다시 하지 않아도 되는 편리함도 누릴 수 있음

쿠키 변조 공격

- 웹사이트의 쿠키를 조작해서 사용자를 속이는 공격
- 쿠키를 조작해 누군가가 사용자의 이름표를 바꾸는 것과 비슷

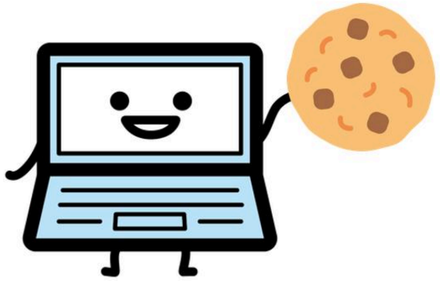
예시

- 인터넷 쇼핑을 할 때 쿠키가 앨리스를 기억하기 위해 쿠키를 사용
- 해커가 그 쿠키를 조작해서 앨리스에 대한 계정 정보를 훔칠 수 있음



나는 앨리스야
내 id와 pwd는 alice/1234야

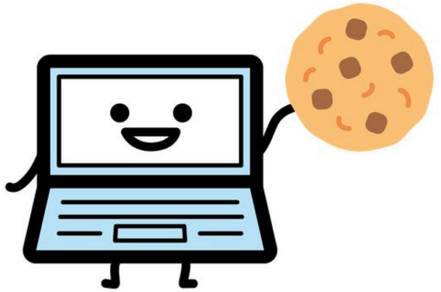
응! 너의 쿠키 값은 qwer로
기억해둘게!



쿠키값 탈취
쿠키값: qwer

내 쿠키값은 qwer이야

앨리스구나!
자, 로그인 시켜줄게



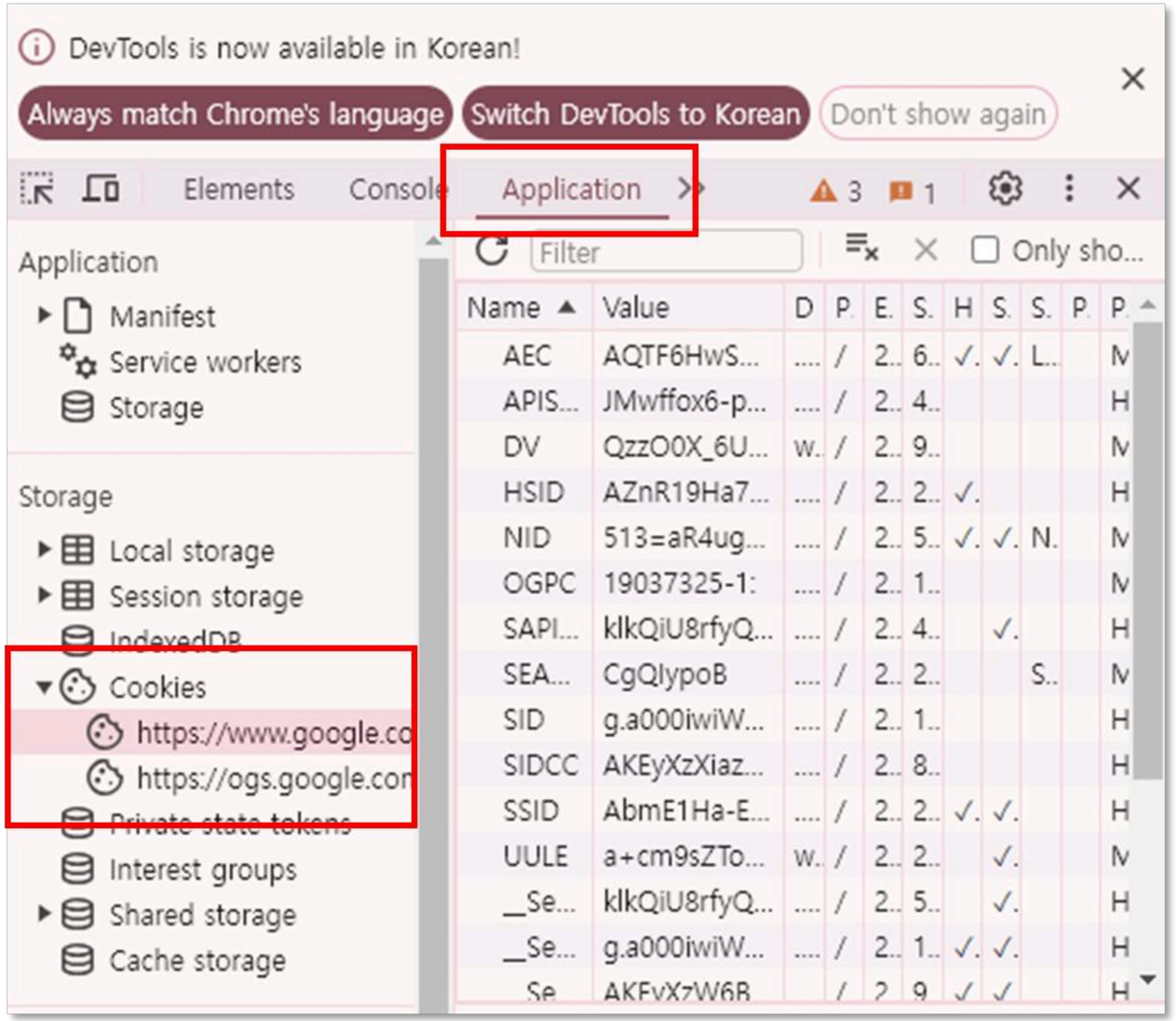
쿠키 변조 실습

RANK	NAME	Hit
1	disso1p1	34 / 100
2	jihun0926	26 / 100
3	ykpak0109	24 / 100
4	acting1	21 / 100
5	zzzzzzzzzz	18 / 100
6	kktt	16 / 100
7	adsbh7	16 / 100
8	sinhw0006	16 / 100
9	hy3on	15 / 100
10	asdd	13 / 100
11	geunyeong	13 / 100
12	jindori96	12 / 100
13	shionista	12 / 100
14	sunny0020	11 / 100
15	wmqkq123	11 / 100
16	dakeee	11 / 100
17	smj100394	11 / 100
18	ynokim	11 / 100
19	s3un9s00n	11 / 100
20	mn961129	10 / 100
21	g0pher	10 / 100
22	simub98	10 / 100
23	seangG	10 / 100
24	tarantu39	10 / 100
25	qwopasklz	10 / 100
26	dlwhdrnr2684	10 / 100
27	raulojeda22	10 / 100
28	hanch7274	10 / 100
29	rlawjdals	9 / 100
30	ziskar2	9 / 100

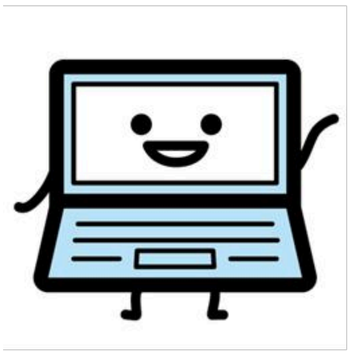
쿠키를 확인해 보는 방법 중 하나!

- 브라우저의 '개발자도구 ' 에 들어가면 볼 수 있다.
- 키보드 맨 위쪽의 ‘F12’ 또는 마우스 우클릭 → 검사

2. 맨 위 상단의 Application(어플리케이션)
3. 왼쪽에 Cookies
4. Cookies를 펼치면 쿠키 값들이 보임



파라미터 변조란?



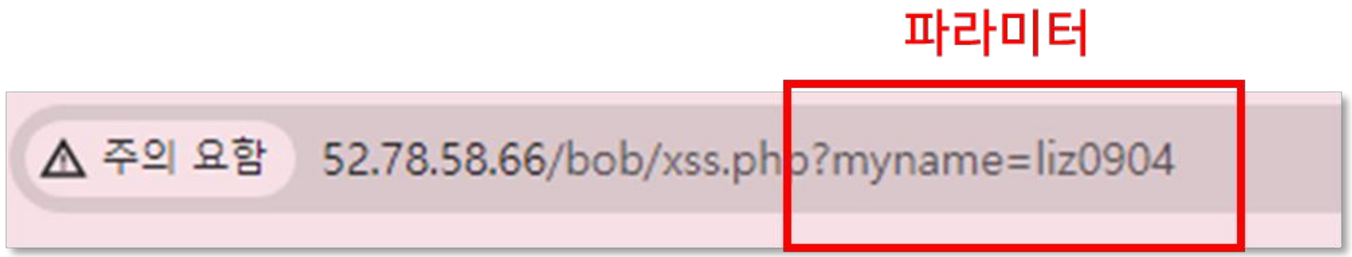
□ + □ = □

What is ‘파라미터 변조'?

- '불충분한 인가 ' 라고도 불림
- 단순하지만 매우 강력한 해킹 공격
- 원래 설정되어 있는 값을 변경하는 걸 의미
- 게임 핵에서도 파라미터 변조를 자주 사용함
- 예) 게임 해킥터의 속도 변경, 다른 사용자의 계정 접속 등

파라미터란?

- 함수나 명령어에 전달되는 값
- 함수나 명령어가 실행될 때 사용되는 데이터



□ + □ = □

1 + 2 = 3

1 + 2 = 3

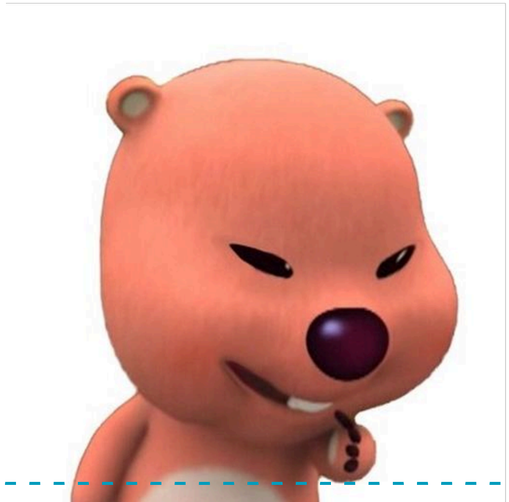
파라미터 파라미터 결과 값

```
user_speed =
```

파라미터 이름

```
user_speed = 80
```

파라미터 이름 파라미터 값



파라미터 변조 발생

```
user_speed = 100000
```

파라미터 이름 파라미터 값

파라미터 변조 사례 – 게임 핵 제작