



# Parameter

# 목 차

---



## 01. Parameter란?



## 02. Parameter 전송 방식

*Get 방식 & Post 방식*



## 03.Parameter변조 공격



## 04. 대응 방안



## 05.Write Up

*Parameter 변조 공격 문제 풀이*



## Q&A



# 01. Parameter란?

## 01. Parameter란?

---

파라미터란?

= 매개변수

컴퓨터 프로그래밍에서 파라미터(parameter, 매개변수)란 변수의 특별한 한 종류로서, 함수와 같은 입력값으로 제공되는 여러 데이터 중 하나를 가리키기 위해 사용된다.

입력 값에 제공되는 여러 데이터들을 전달인자라고 부른다.

```
int sum(int x, int y)  
{  
    return x+y;  
}
```



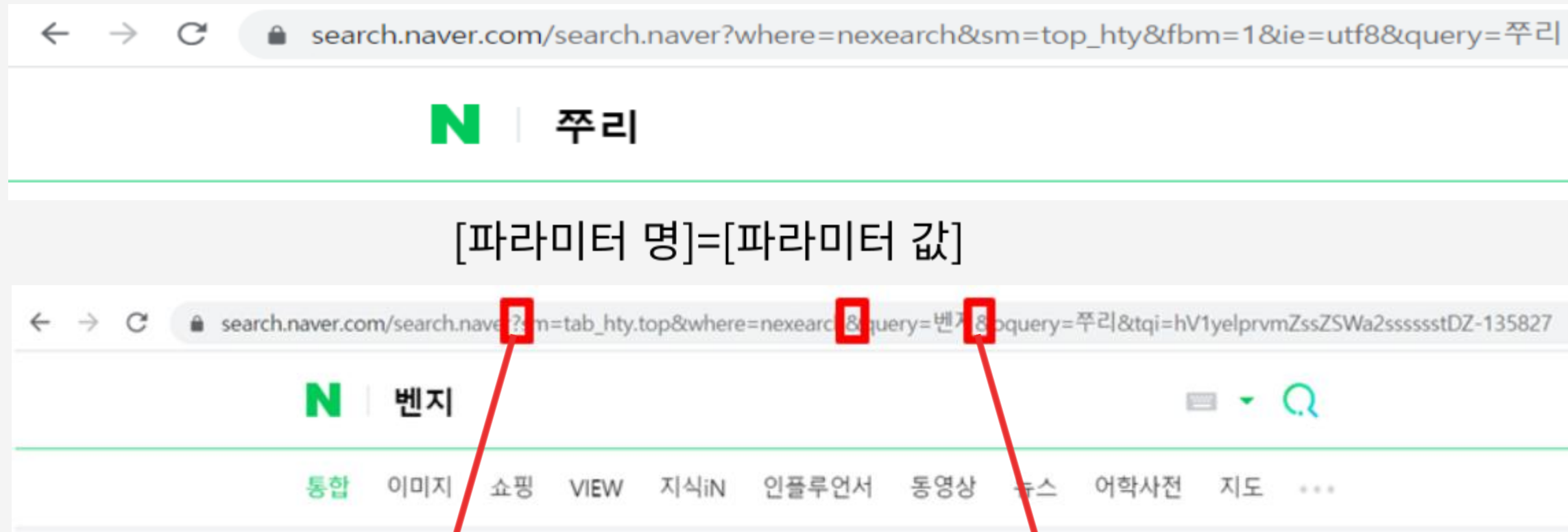
## 02. Parameter 전송 방식

*Get 방식 & Post 방식*

---

## 02. Parameter 전송

Get 방식



이후부터 URL 파라미터 영역

파라미터가 여러 개 일 때 '&'로 이어줌



## 02. Parameter 전송

### Post 방식

요청 라인	상태 라인
<u>헤더</u>	헤더
공백	공백
바디	바디

HTTP 패킷의 헤더에 Body의 콘텐츠 타입을 명시하는 Content-Type 포함하여야 한다.

Get 방식과 달리 http 패킷 body에 데이터가 담겨져 전송



02. Parameter 전송

| 차이점 정리

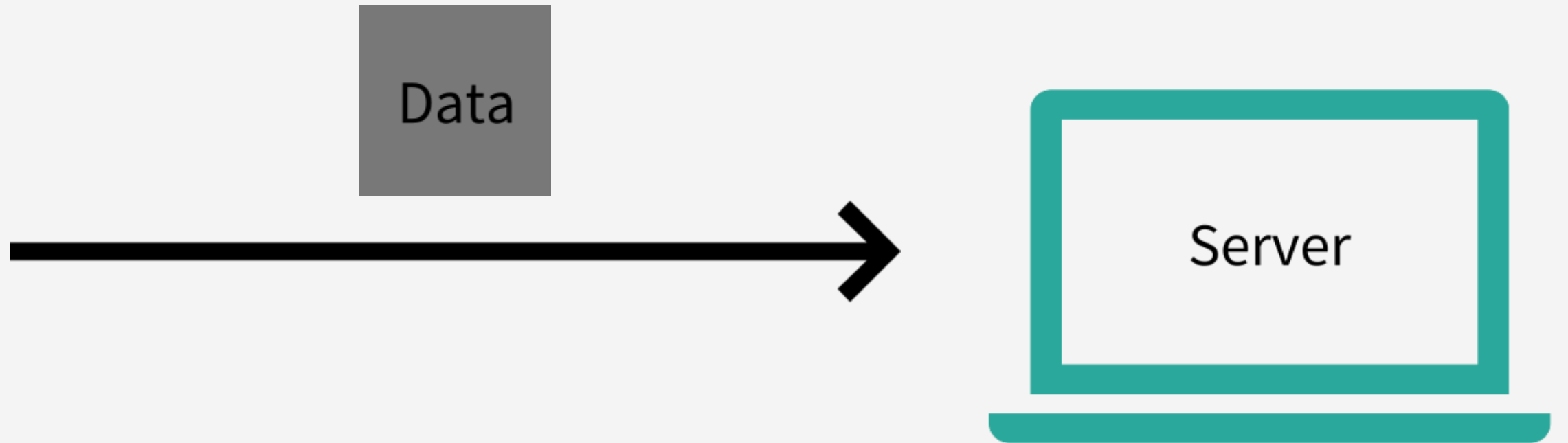
	Get 방식	Post 방식
데이터, 파라미터 전송 방법	URL에 붙여 전송	Body에 넣어 전송
데이터 길이 제한	유한(256 byte)	무한
전송 속도	빠름	느림
보안성	낮다	높다





## 03. Parameter변조 공격

### 03. Parameter변조 공격 | Parameter 변조 공격이란?

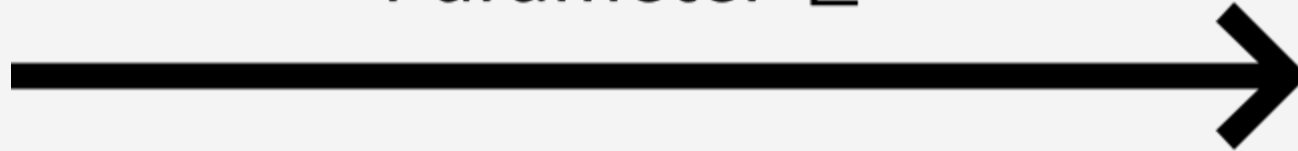


### 03. Parameter변조 공격 | Parameter 변조 공격이란?

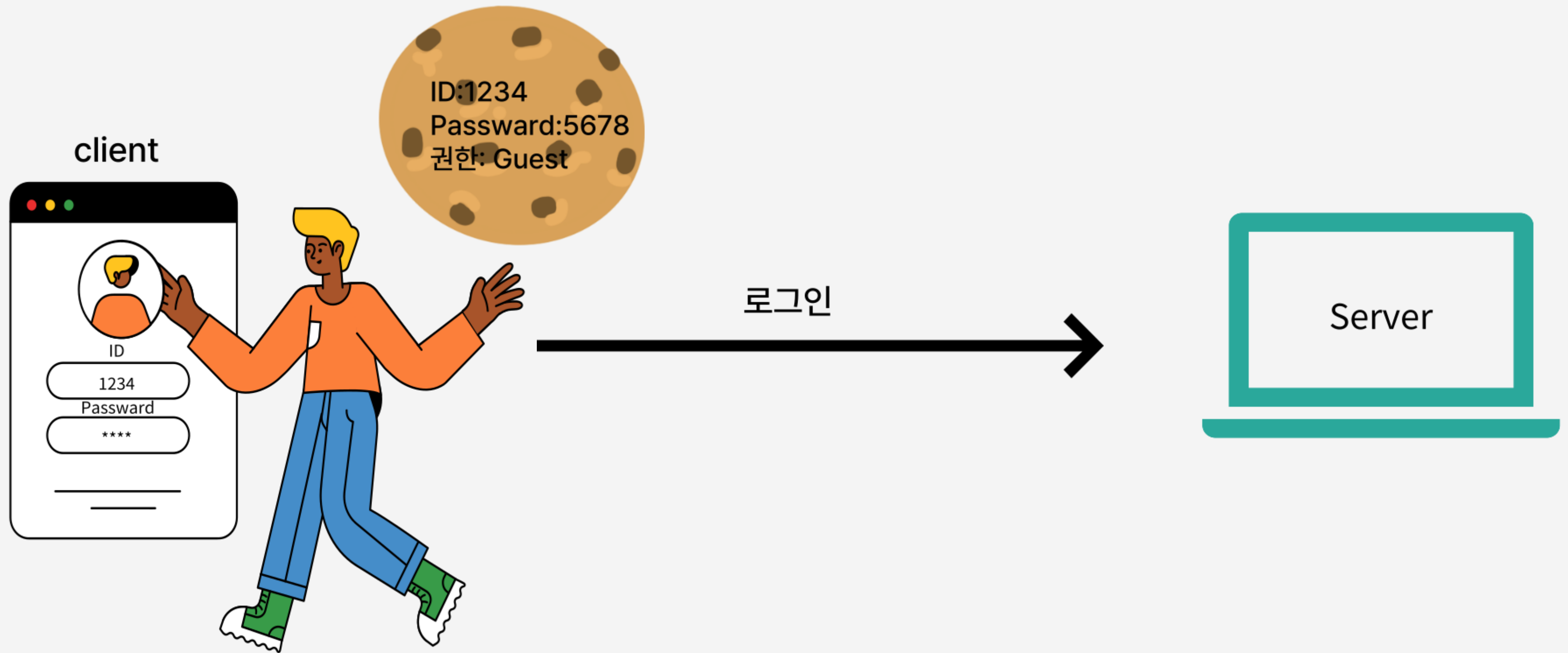


공격자

Parameter 변조

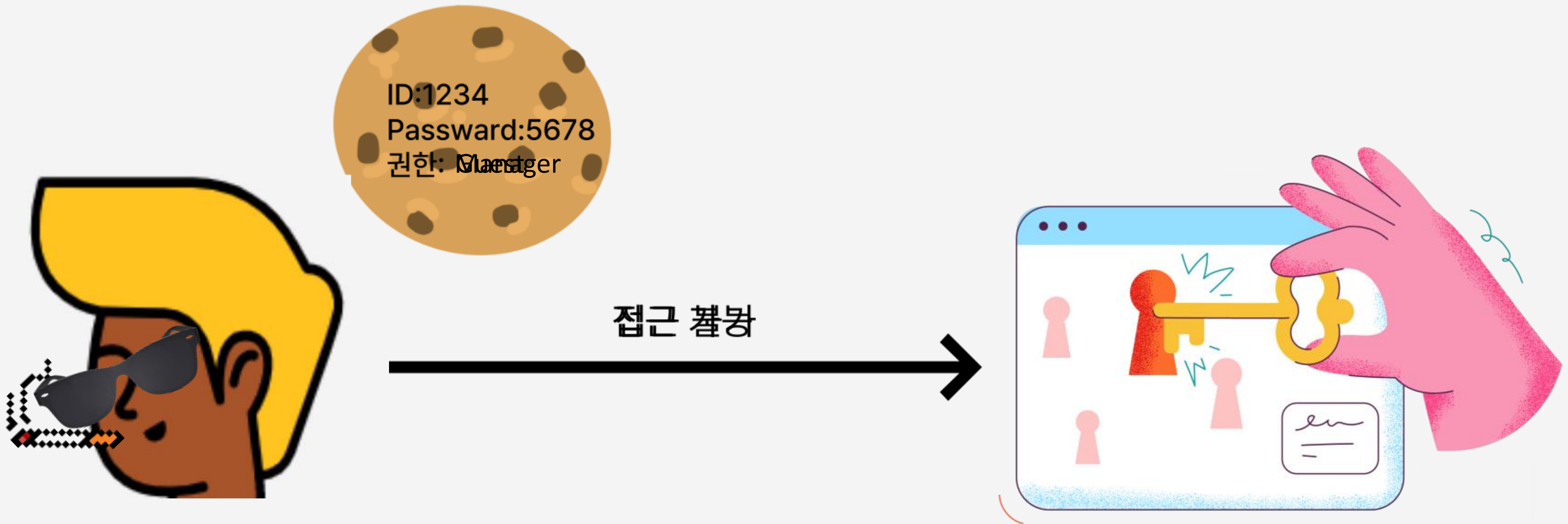


### 03. Parameter변조 공격 | Parameter 변조 공격이란?



### 03.Parameter변조 공격

Parameter 변조 공격이란?



다양한 파라미터 값을 변조하여 부당한 이득을 취하거나 접근 권한을 상승 시켜 정보를 조회/변경할 수 있다.



## **04. 대응 방안**

# 04. 대응 방안

## 안전한 쿠키 사용

Response Headers

view source

Cache-Control

no-store, no-cache, must-revalidate, post-check=0, pre-check=0

Connection

Keep-Alive

Content-Encoding

gzip

Content-Length

83

Content-Type

text/html

Date

Sun, 17 Mar 2013 20:07:52 GMT

Expires

Thu, 19 Nov 1981 08:52:00 GMT

Keep-Alive

timeout=15, max=100

Pragma

no-cache

Server

Apache/2.2.16 (Debian)

Set-Cookie

bb07750878258c3d0b8379b49daf0193=5x4xRKLIKSSkeedej1zBg4rtPC8v9XRSTZihnqLDr6J%2FvV27%2Fx6le%2F%2Fts0XHMObtldqsrrFupnDOT2faKN33N1sji48e7p

Vary

Accept-Encoding

X-Powered-By

PHP/5.3.3-7+squeezel5

암호화된 쿠키 값

## 개인정보 페이지 본인 인증 적용

회원정보 조회 페이지

고객명	홍길동
아이디	hongkdong
생년월일	1990.09.10

비밀번호를 입력해 주세요.

회원님의 정보를 안전하게 보호하기 위해 비밀번호를 다시 한번 확인합니다.

아이디

hongkdong

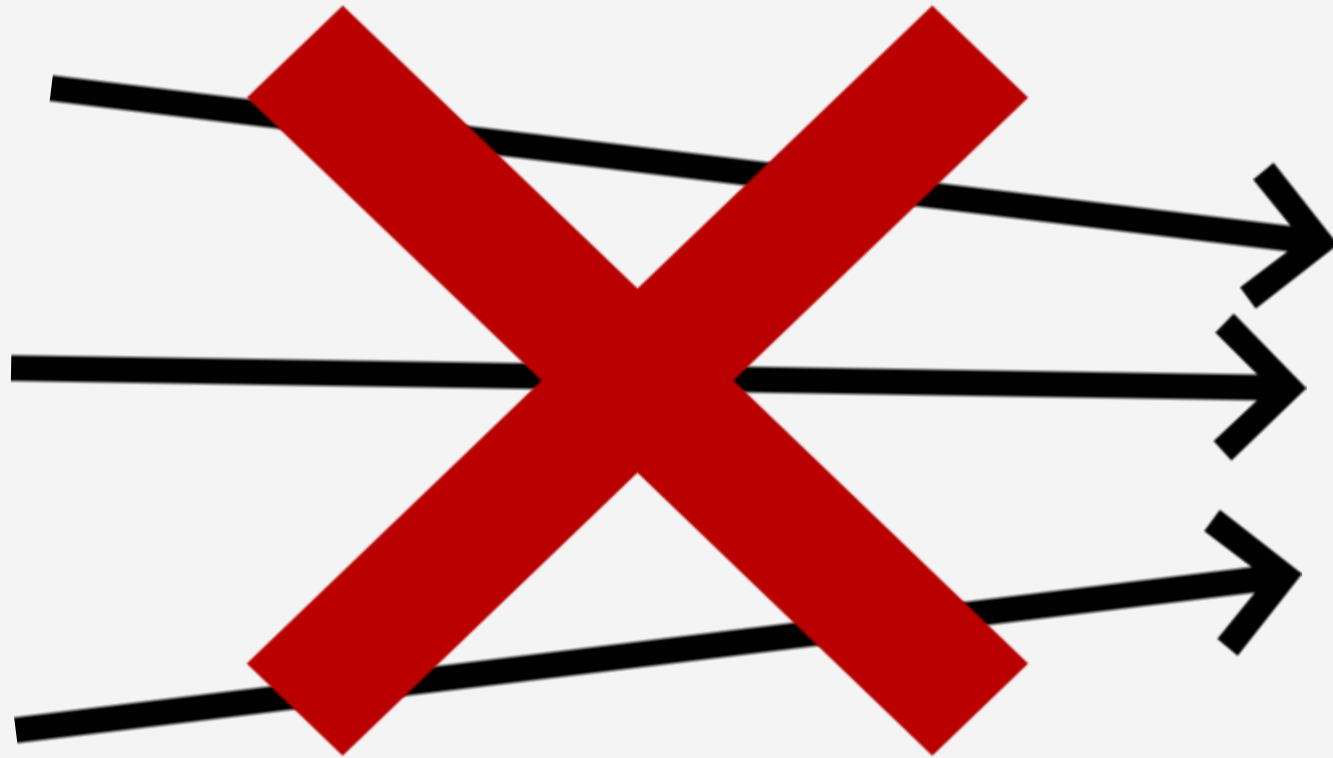
비밀번호

확인

## 04. 대응 방안

---

특정 아이피에서 계속되는 수상한 접근 시도 차단







# 05. Write Up

*Parameter 변조 공격 문제 풀이*


## 05. Write Up

| webhacking.kr 42번

### 문제 확인

no	subject	file
2	test	test.txt <a href="#">[download]</a>
1	read me	flag.docx <a href="#">[download]</a>

### test.txt 파일 내용

 test.txt - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

test123  
qwer123  
asdf234

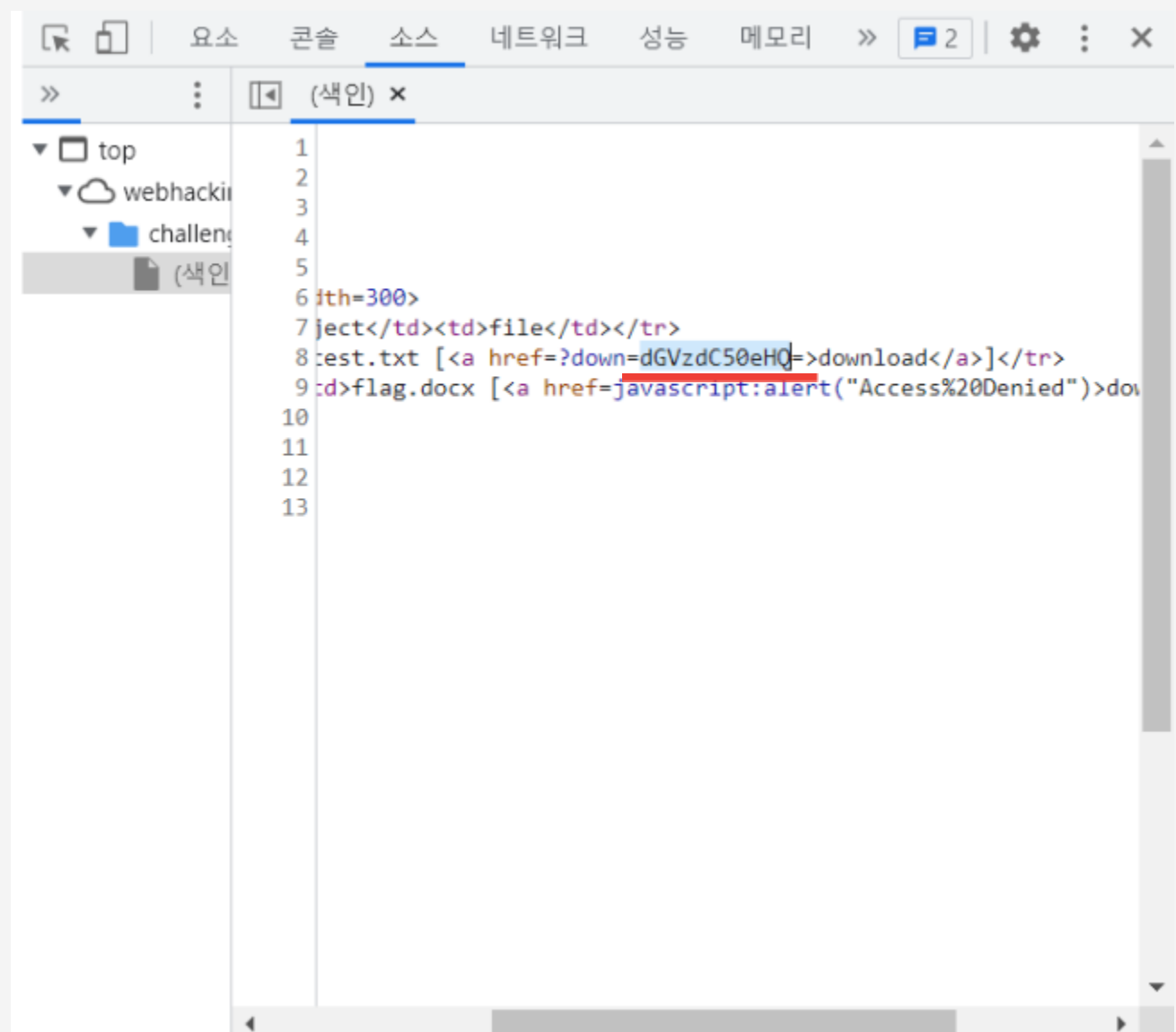
### flag.docx 파일 다운로드 시도

webhacking.kr의 메시지

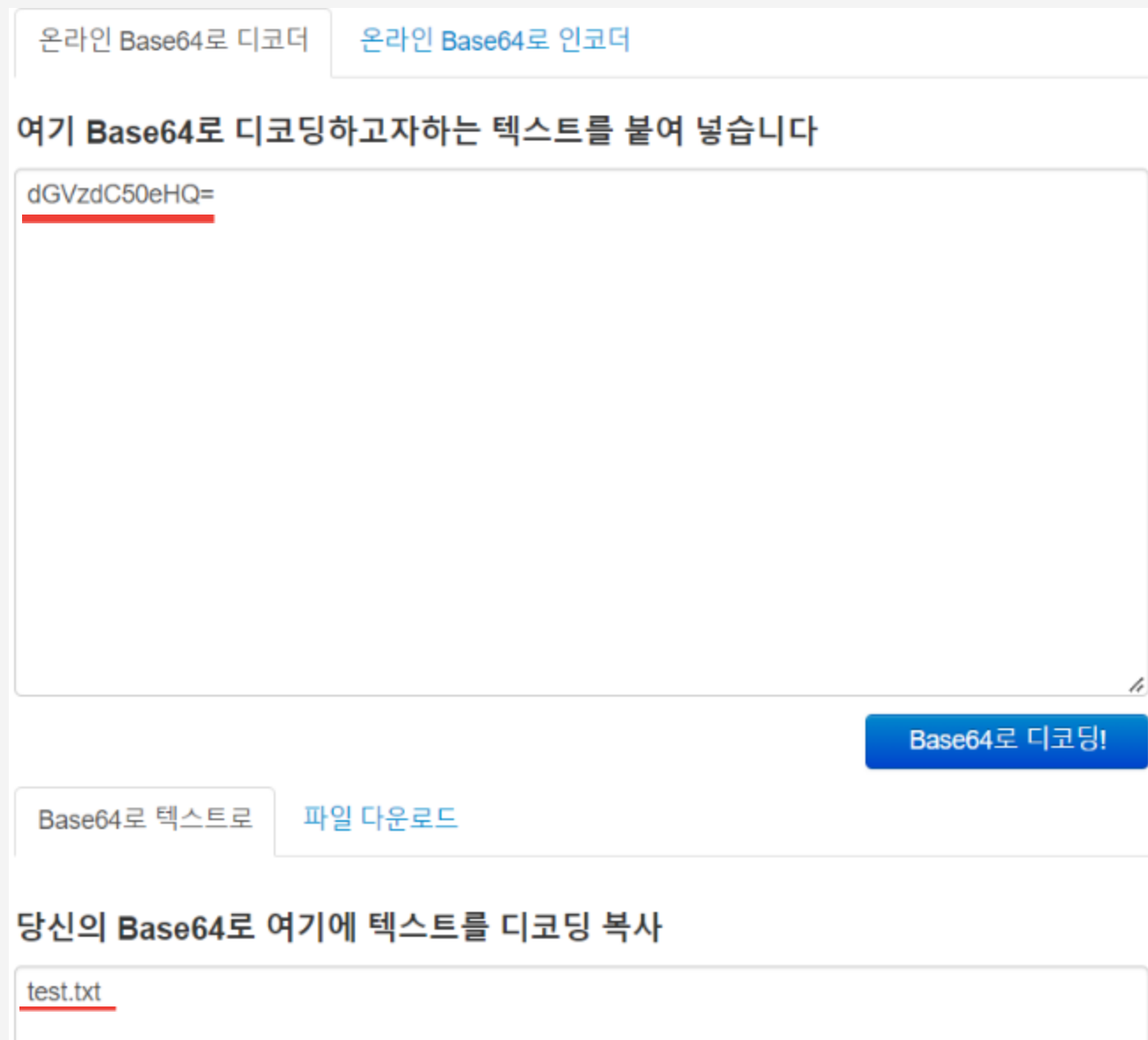
Access Denied

확인

## 개발자 모드(F12)를 통한 소스코드 확인



## 디코딩 결과



## 05. Write Up

| webhacking.kr 42번

### flag.docx 인코딩

flag.docx

Base64로 인코딩!

여기 Base64로 인코딩 된 텍스트를 복사 :

ZmxhZy5kb2N4

 <https://webhacking.kr/challenge/web-20/?down=ZmxhZy5kb2N4>

 <https://webhacking.kr/challenge/web-20/?down=ZmxhZy5kb2N4>

flag!

FLAG{very\_difficult\_to\_think\_up\_text\_of\_the\_flag}

### 문제 확인

**Password is 8**

### 동체시력 테스트

Auth Flag

FLAG{8204ede801}

invalid flag

Close

Auth

### 소스코드 확인

```
function answer(i){  
  x.open('GET', '?m='+i, false) //get 방식  
  x.send(null);  
  aview.innerHTML=x.responseText;  
  i++;  
  if(x.responseText) setTimeout("answer(++i)", 20); //서버에서 전달받은 데이터가 있다면 20밀리세컨드 후 재귀함수 호출  
  if(x.responseText=="") aview.innerHTML="?"; //서버에서 전달받은 데이터가 공백, 즉 끝이라면 aview에 ?를 출력  
}  
setTimeout("answer(0)", 1000);
```

## 05. Write Up

| webhacking.kr 54번

콘솔에서 소스코드 수정

```
function answer(i){  
  x.open('GET','?m='+i,false);  
  x.send(null);  
  aview.innerHTML+=x.responseText;  
  i++;  
  if(x.responseText) setTimeout("answer("+i+")",20);  
  if(x.responseText=="") aview.innerHTML+= "?";  
}  
setTimeout("answer(0)",1000);
```

flag!

Password is ?FLAG{a7981201c48d0ece288afd01ca43c55b}?

---

# Q&A

*Thank you*

---