

서울아이티고등학교 해킹방어대회 Writeup

Introduce

- Author : 김한솔 (Circler)
- E-Mail : dkdlelgksthf@gmail.com
- Blog : [naver blog](#)
- Date : 2017-09-24

Review

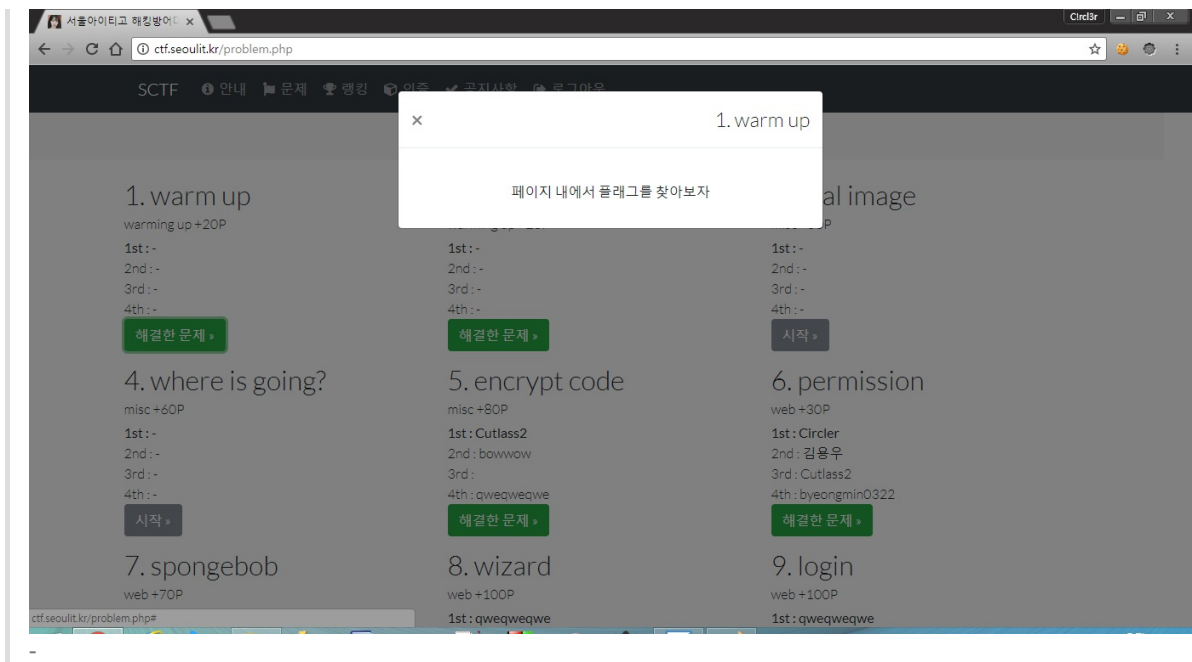
- 생각했던 난이도보다 쉬웠다.
- 초반에 4등까지 올라갔지만 후반에 리버싱을 풀지 못해서 엄청 내려갔다.
리버싱을 공부하자

Solve

- 문제의 번호순서대로 작성하고 번호는 문제의 번호이다.

1. warm up

- 플래그를 찾아야한다.



- 페이지소스코드를 보자.

```
610 <div class="container">
611 <div id="message" style="text-align:center"></div>
612 <div style="text-align: center;">
613 <input type="text" class="flag_input" name="flag" placeholder="정답을 입력 해주세요."
614 required>&nbsp;  <input type="button" id="input_flag" class="btn btn-secondary" value="전송">
615 </div>
616 </div>
617 </form>
618 </div>
619 </div>
620 </div>
621 </div>
622 <!--script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script-->
623 <script src="http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
624 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-alpha.5/js/bootstrap.min.js"
625 integrity="sha384-VMLz127B6a9D4B4B3R9VHj172pk6B2-4C009hzyeWEeJ01e6izqLForm"
626 crossorigin="anonymous"></script>
627 <script src="frontend/js/bootstrap.min.js"></script>
628 <script src="frontend/js/sign-modal.js"></script>
629 <script src="frontend/js/page_min.js"></script>
630 <script src="frontend/js/welcome.js"></script>
631 <script src="frontend/js/welcome.js"></script>
632 <script src="frontend/js/welcome.js"></script>
633 </script>
634 <script>
635 (function(i,s,o,g,r,a,m){
636 i['GoogleAnalyticsObject']=r;
637 i[r]=i[r]||function(){
638 (i[r].q=i[r].q||[]).push(arguments)
639 },i[r].l=1*new Date();
640 a=s.createElement(o),
641 m=s.getElementsByTagName(o)[0];
642 a.async=1;
643 a.src=g;
644 m.parentNode.insertBefore(a,m)
645 })(window,document,'script','https://www.google-analytics.com/analytics.js','ga');
646 ga('create','UA-77370571-7','auto');
647 ga('send','pageview');
648 </script>
649 </script>
650 <script src="frontend/js/flag.js"></script>
651 </script>
652 </body>
653 </html>
```

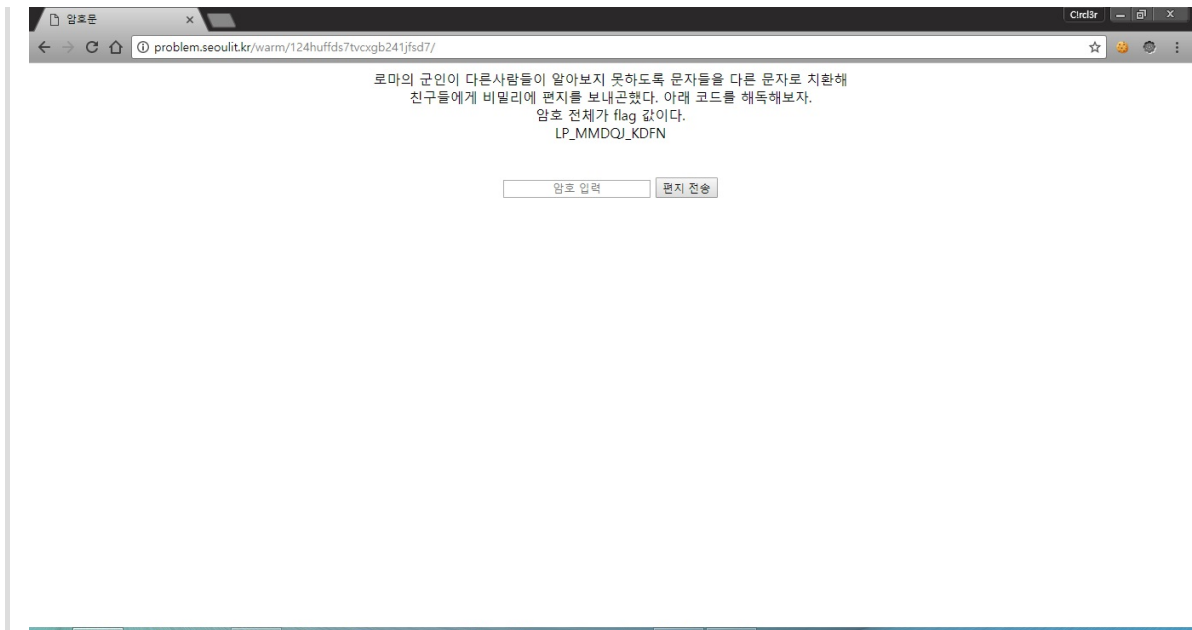
- welcome.js 파일이 있다.

```
// alert("flag is im_jjang_hacker")
```

- 플래그가 있다.

2. warm up

- 시작



- 로마의 군인이 쓰는 암호 = 99% 카이사르 암호



- 복호화를 하고 전송을 하면 플래그가 나온다.
-플래그인증을 하고나서 생각해봤는데 플래그도 암호화 되있는 것 같지만 결국 복호화를 못했다.

5. encrypt code

- 문제를 열면 자바코드를 준다.

배열에다가 숫자를 집어넣고 배열을 또 입력받아서 XOR연산 후 Ruby_is_light와 동일한지 묻는다.

- ```
nyam=[24,20,20,24,0,0,0,0,4,12,6,30,13]
hateRuby="Ruby_is_light"
str=""
for i in range(0,13):
 str+=chr(nyam[i]^ord(hateRuby[i]))
print(str)
```

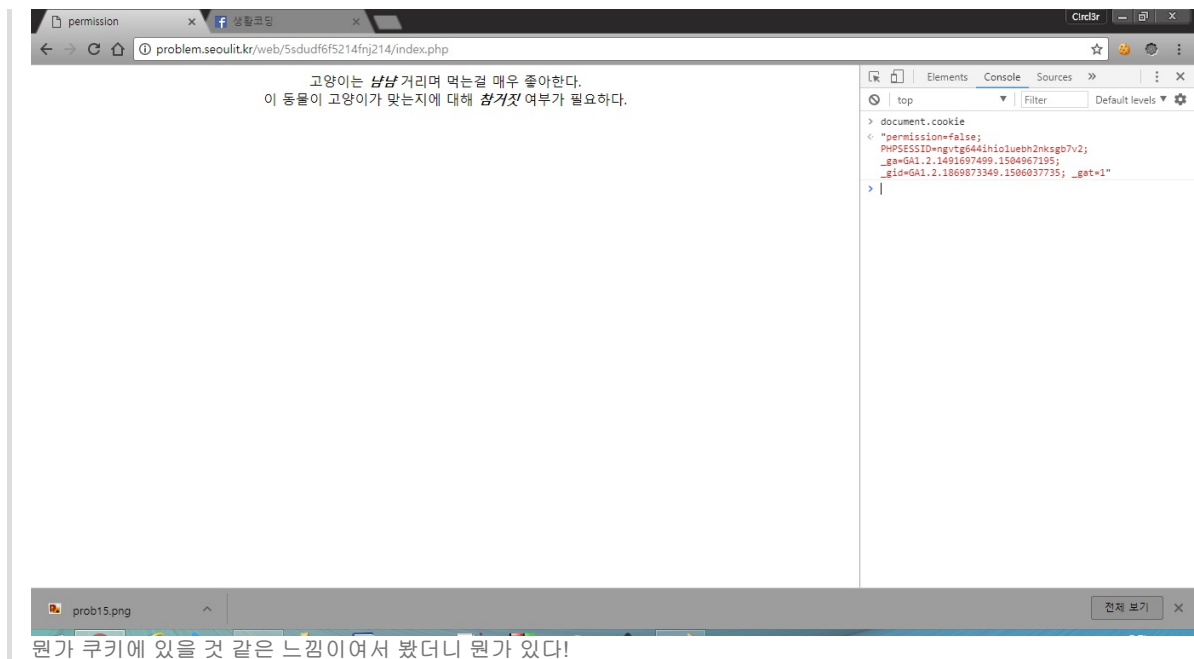


6. permission

해결한 문제 »

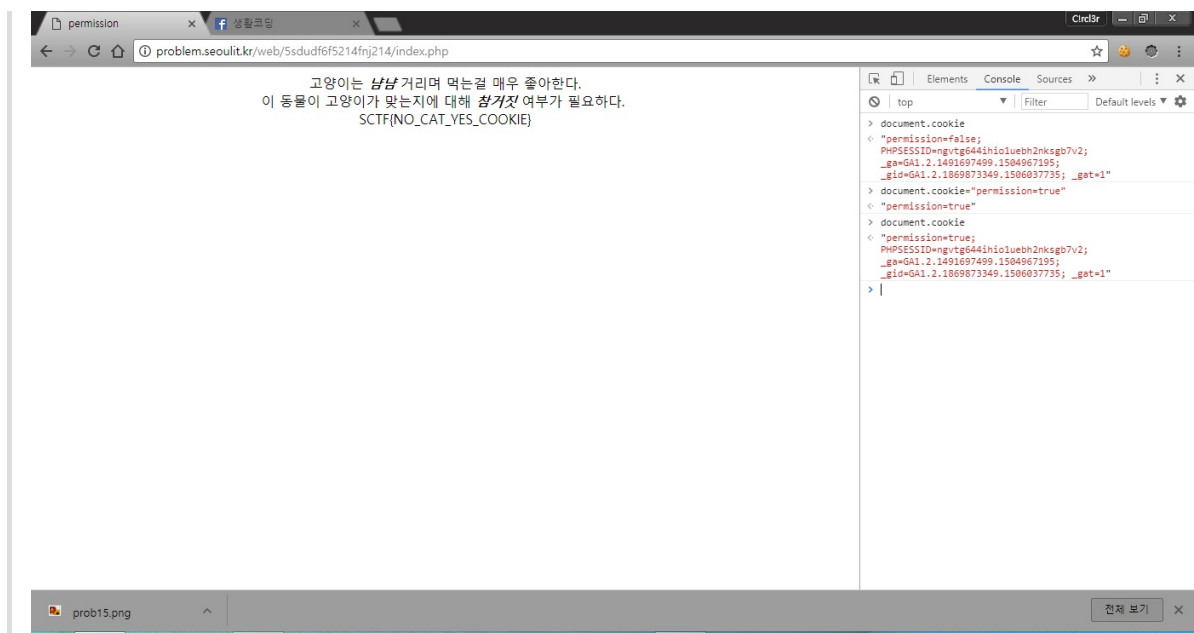
## First Solve!

- 문제를 보면 아무것도 없이 참거짓의 여부를 묻고있다.



뭔가 쿠키에 있을 것 같은 느낌이어서 봤더니 뭔가 있다!

- 자바스크립트로 쿠키를 변경한 다음에 새로고침하면 플래그가 나온다.



## 7. spongebob

## 7. spongebob

web +70P

1st : Circler

2nd : bAoBaP\_hyom

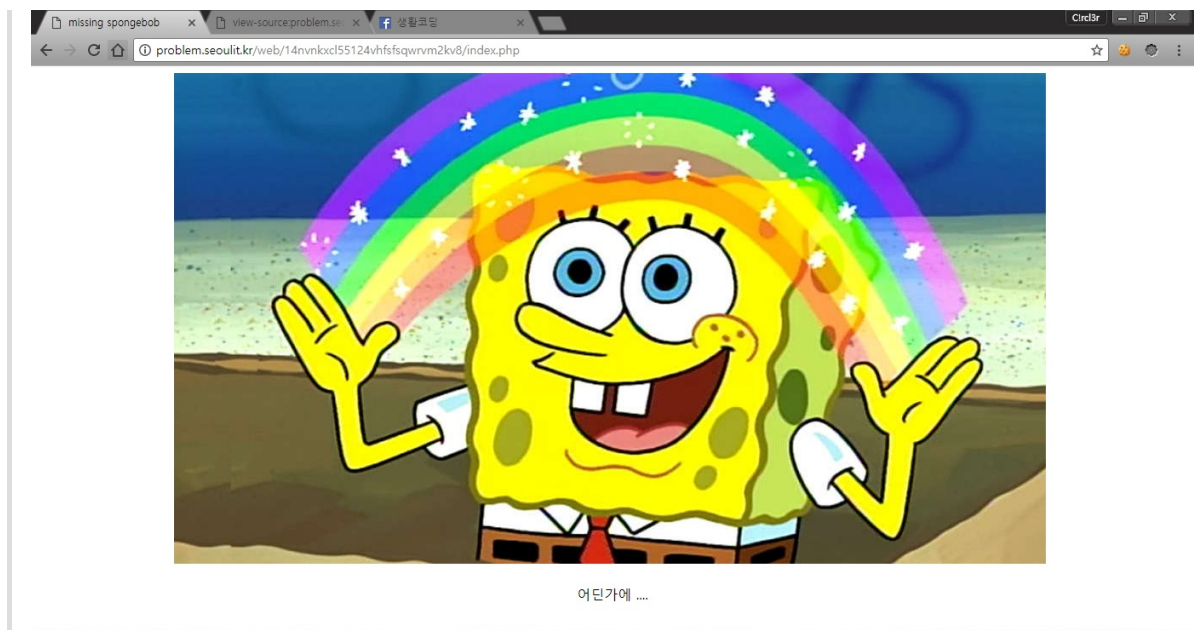
3rd :

4th : D41JUNGOD

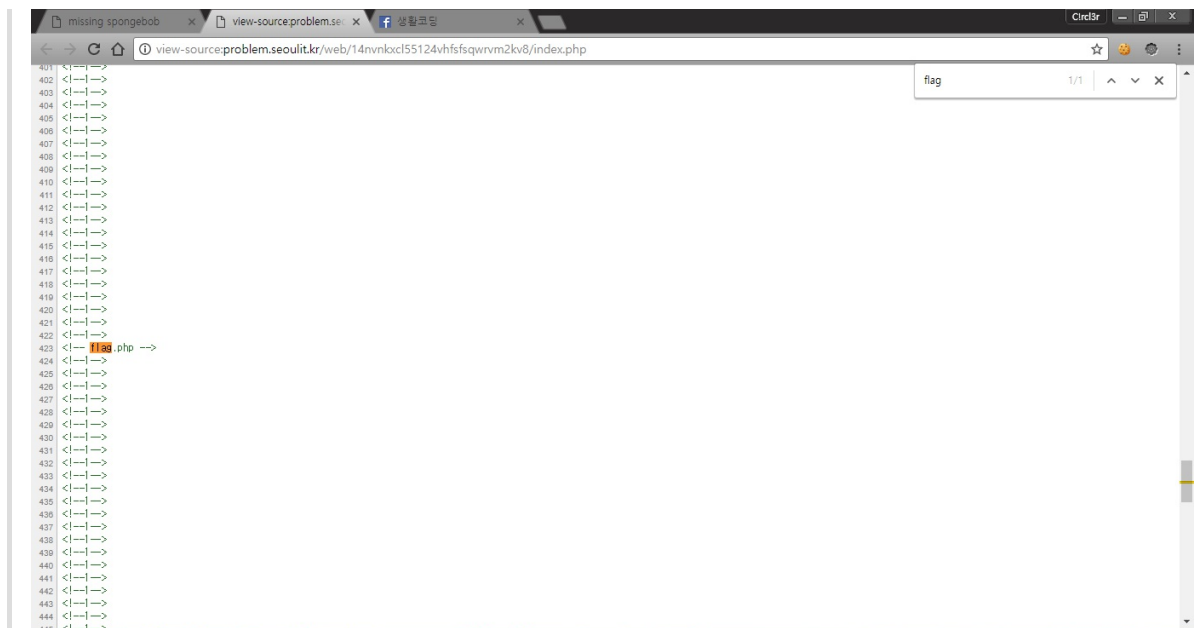
해결한 문제 »

First Solve!

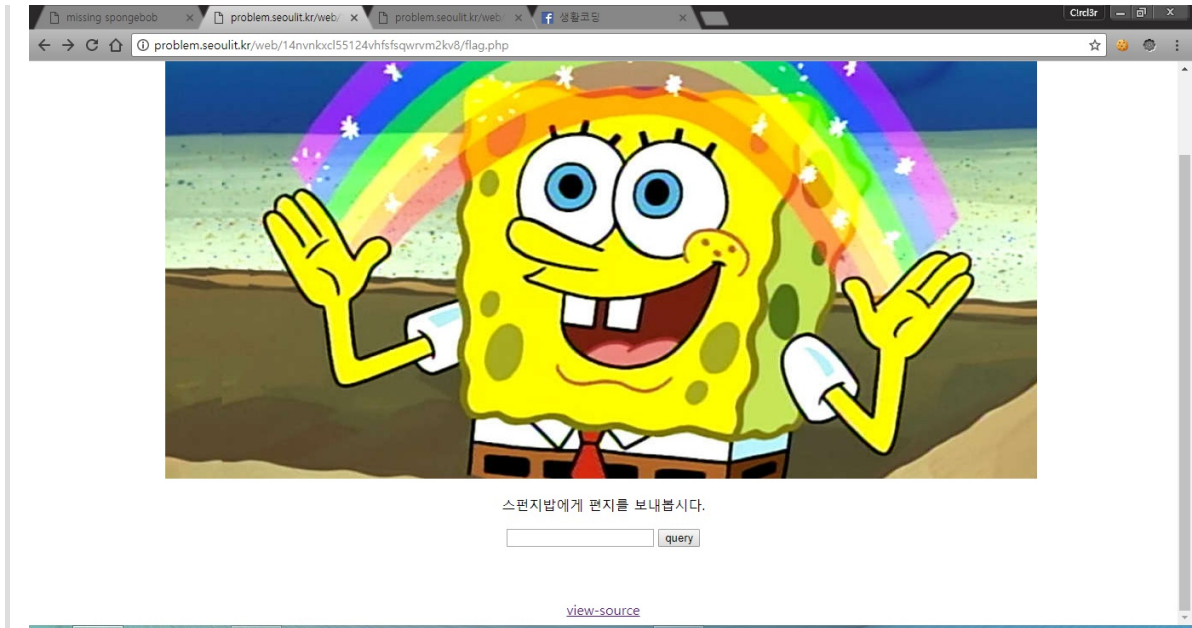
- 스폰지밥이 무지개빛 무지개와 함께 있다.



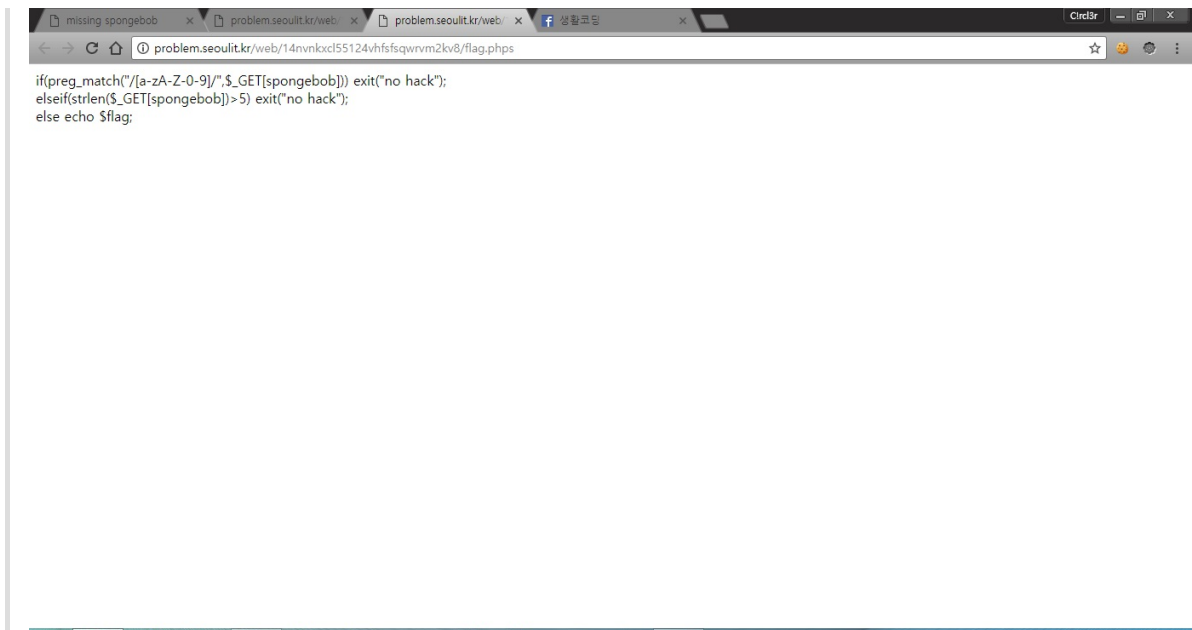
- '어딘가에' 라는 문구가 힌트인 것 같아서 소스코드를 봤더니 어느 페이지가 나온다.



- 들어가면 편지를 보내야한다고 나온다.

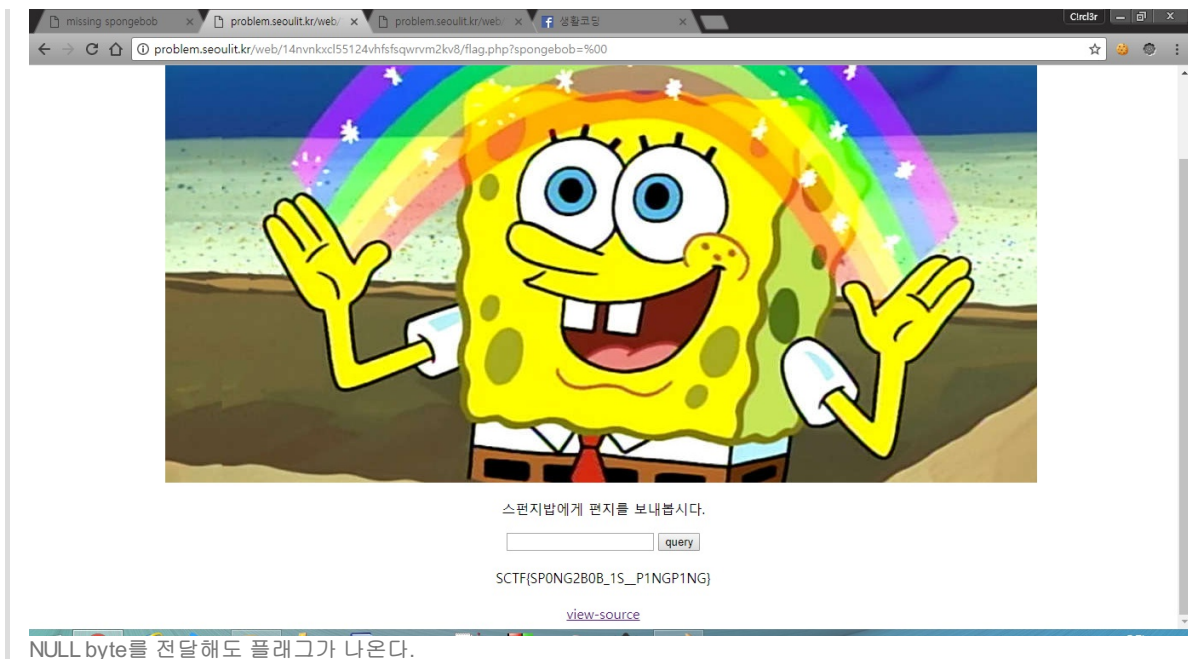


- view-source 있길래 들어가 봤다.



처음 문제를 봤을 때는 이런 텍스트 없었는데 풀고나서 10분 쯤 있다가 나온 것 같다.  
정규식인 [a-zA-Z-0-9]을 이용해서 영문자, 숫자가 GET방식으로 넘어올 경우 필터링한다.  
위 정규식에 걸리지 않는 문자는 "~!@#\$\$%^&\*()\_-= 한글"

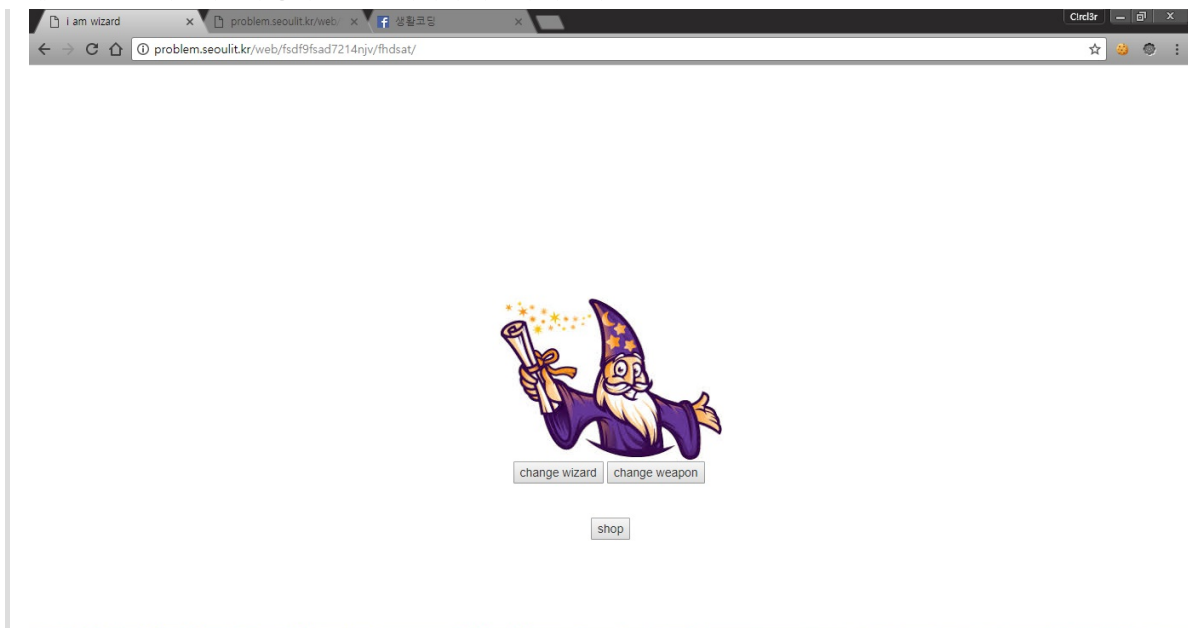




NULL byte를 전달해도 플래그가 나온다.

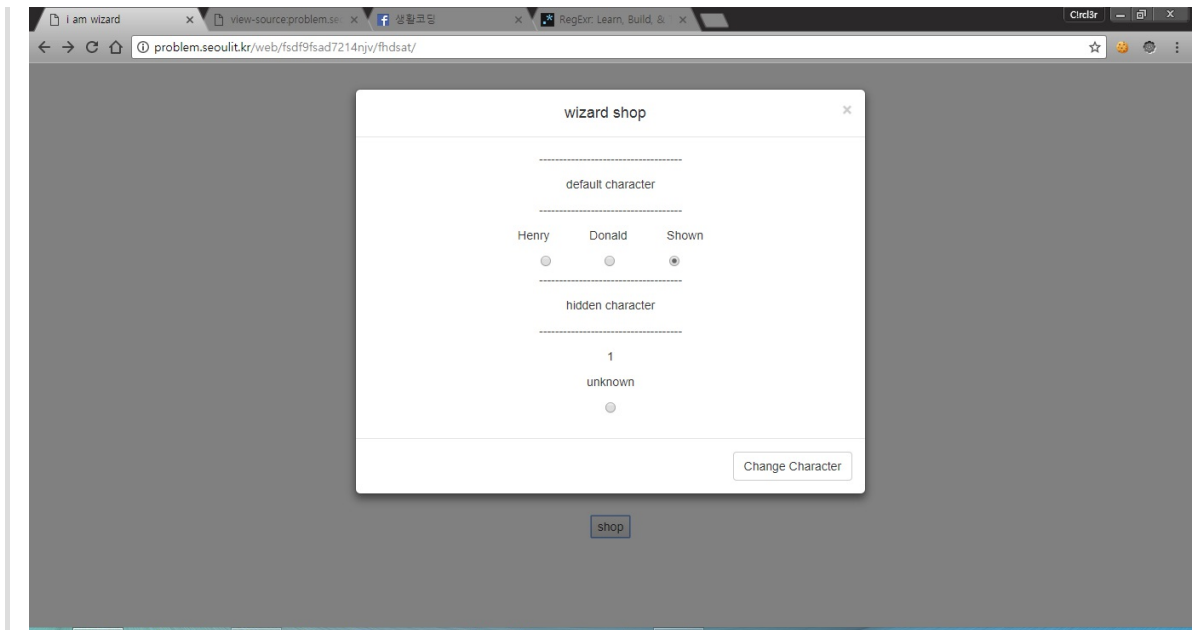
## 8. wizard

- 문제를 보면 웬 마법사로 추정되는 2D 할아버지 사진이 나온다.

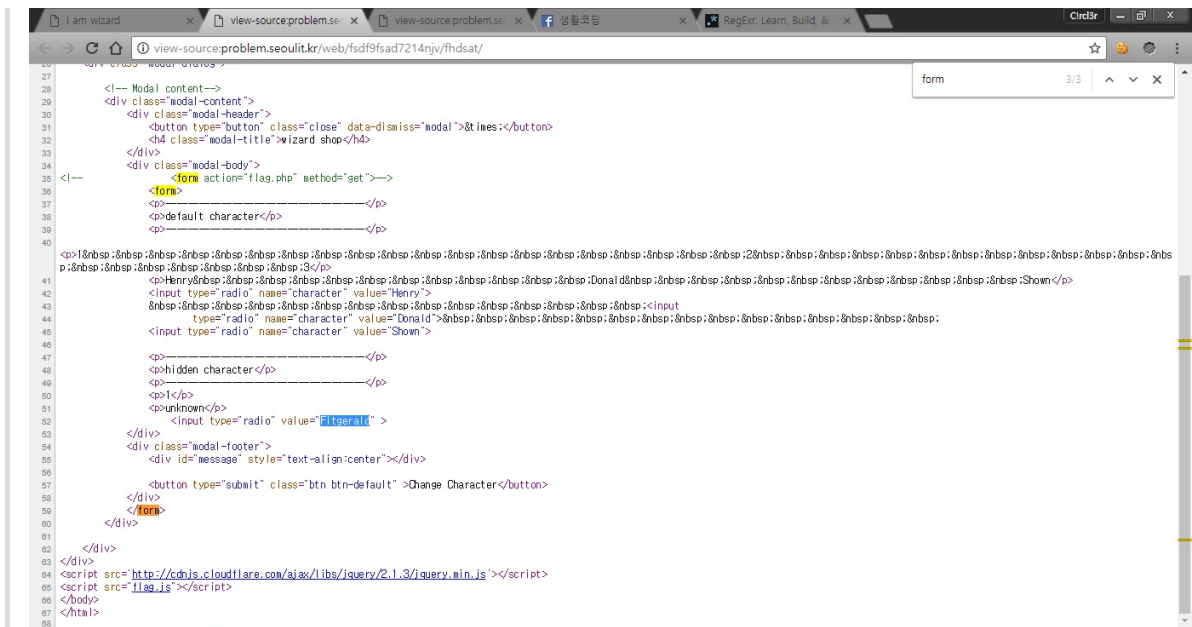


- change wizard 나 change weapon 버튼을 눌러도 아무런 반응이 없지만 shop 버튼을 누르면 뭔가 뜬다.

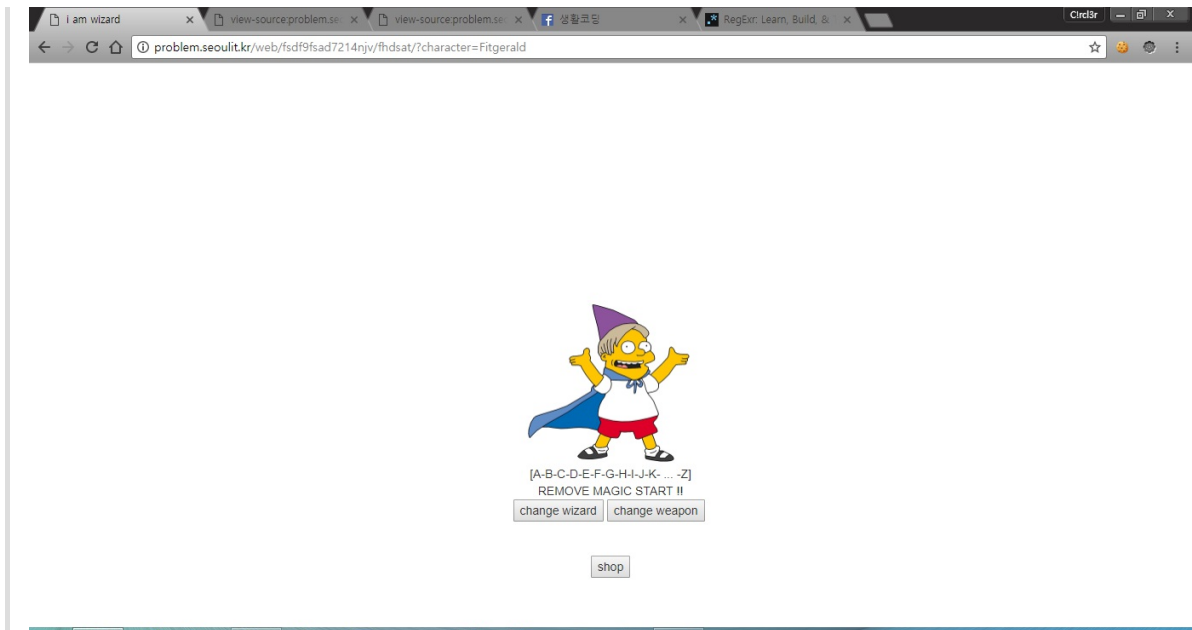




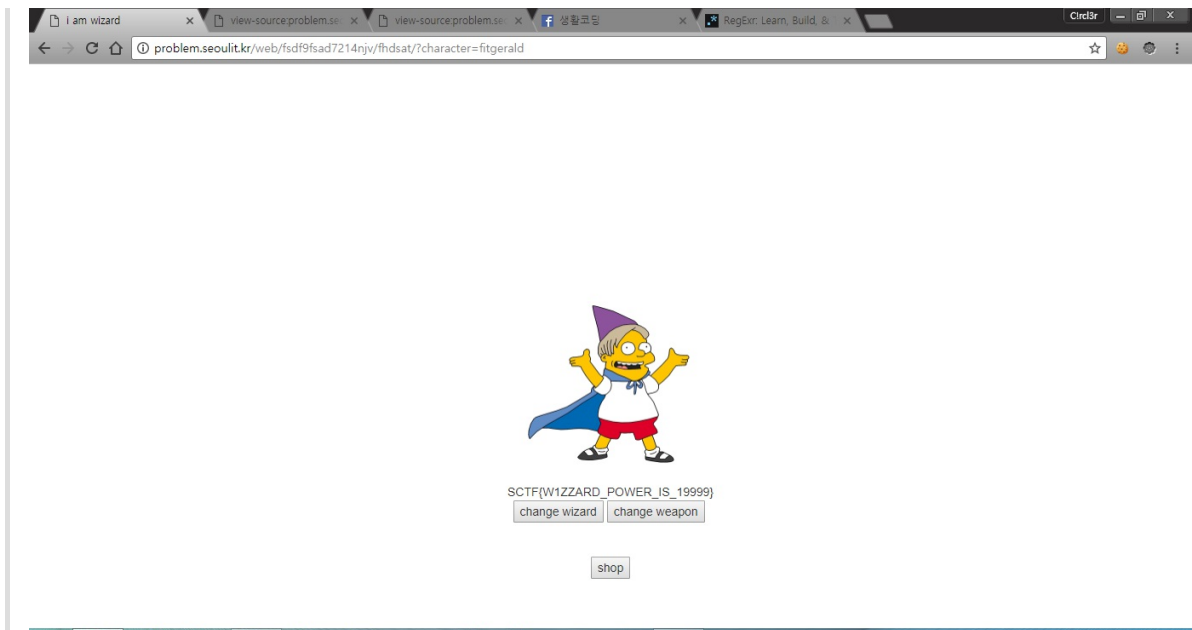
- 코드를 봤더니 unknow n에 value만 가진 input태그가 있었다. (name속성이 없으면 GET으로 값을 받지 못함)



- 일단 value속성이 가진 값을 GET파라미터의 character로 넘겨주면 심슨캐릭터가 뜬다.



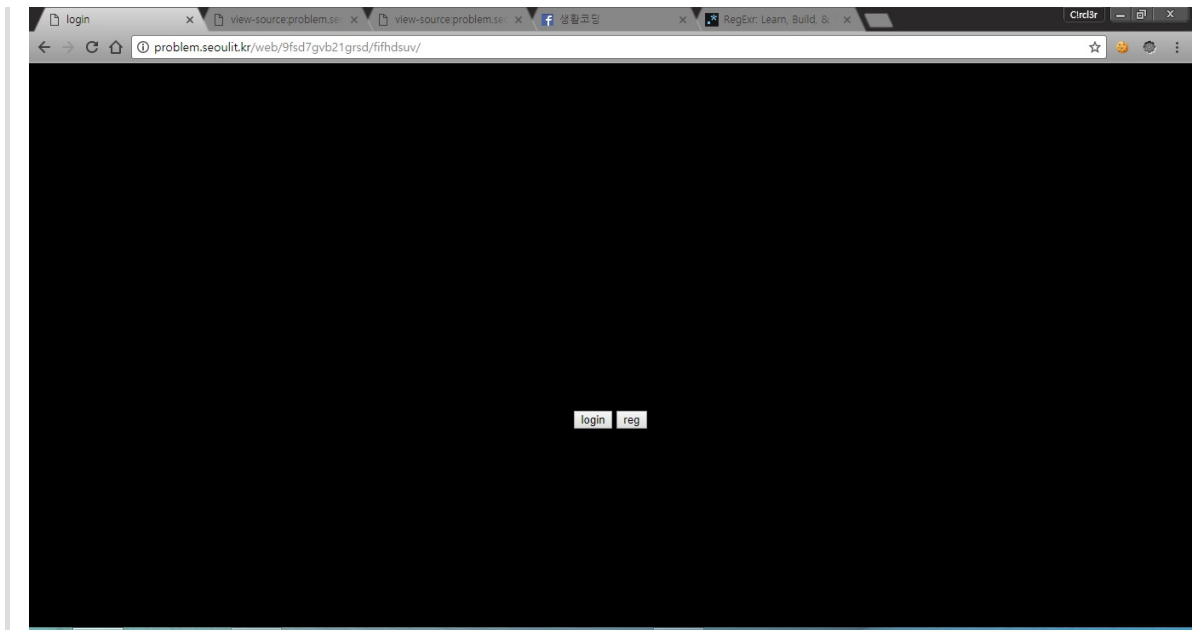
- 이미지에 있는 [A-B-C-D-E-F-G-H-I-J-K-...-Z] 이것이 왜 있는지 한참을 생각한 결과 정규표현식이라고 결론을 내었고 실제로 저 패턴을 ([regexr.com](http://regexr.com)) 사이트에서 테스트를 해보니 실제로 알파벳 대문자만 검색되는 것이었다.



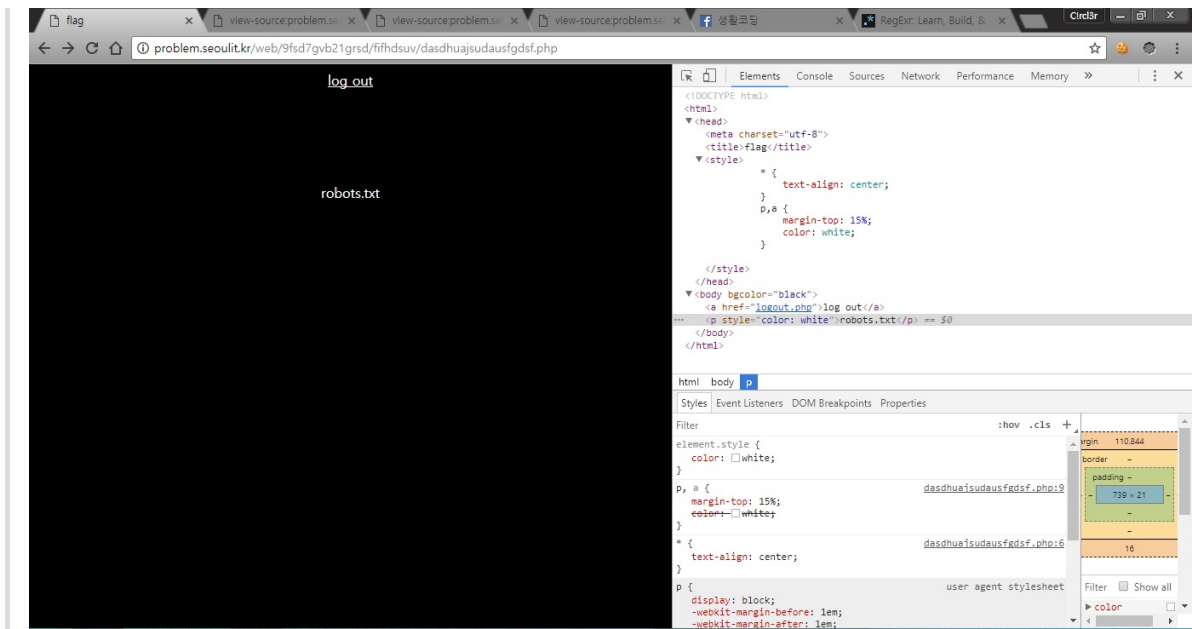
- URL에 있는 Fitzgerald를 fitgerald로 바꿔주면 플래그 획득

## 9. login

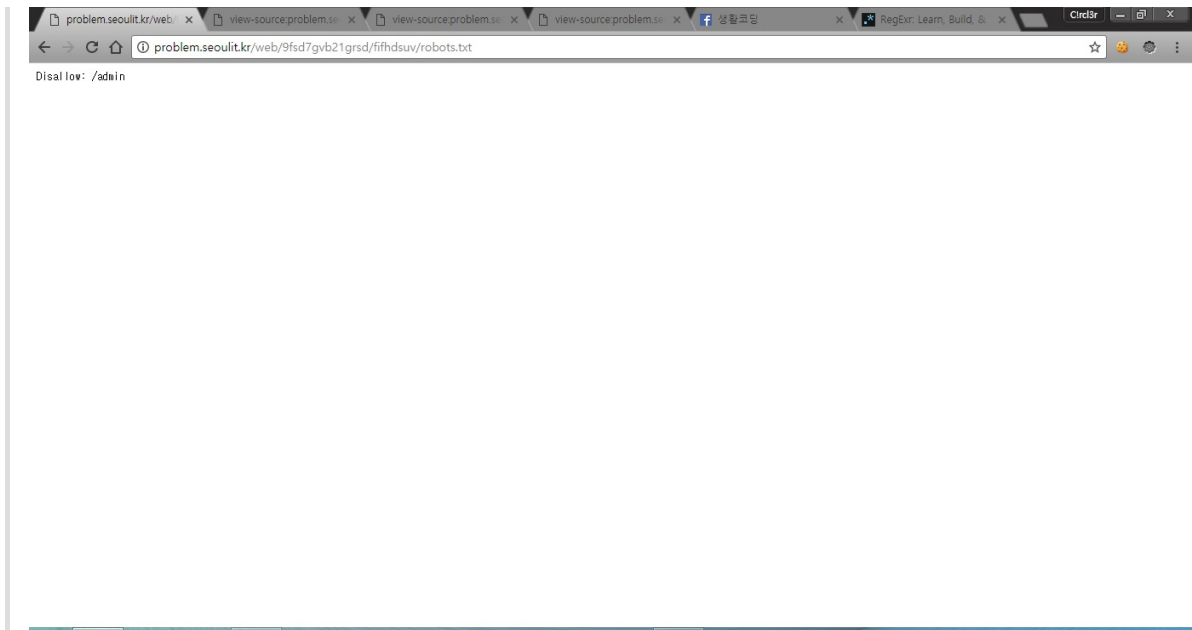
- 우선 문제를 열면 검은 화면이 나온다.



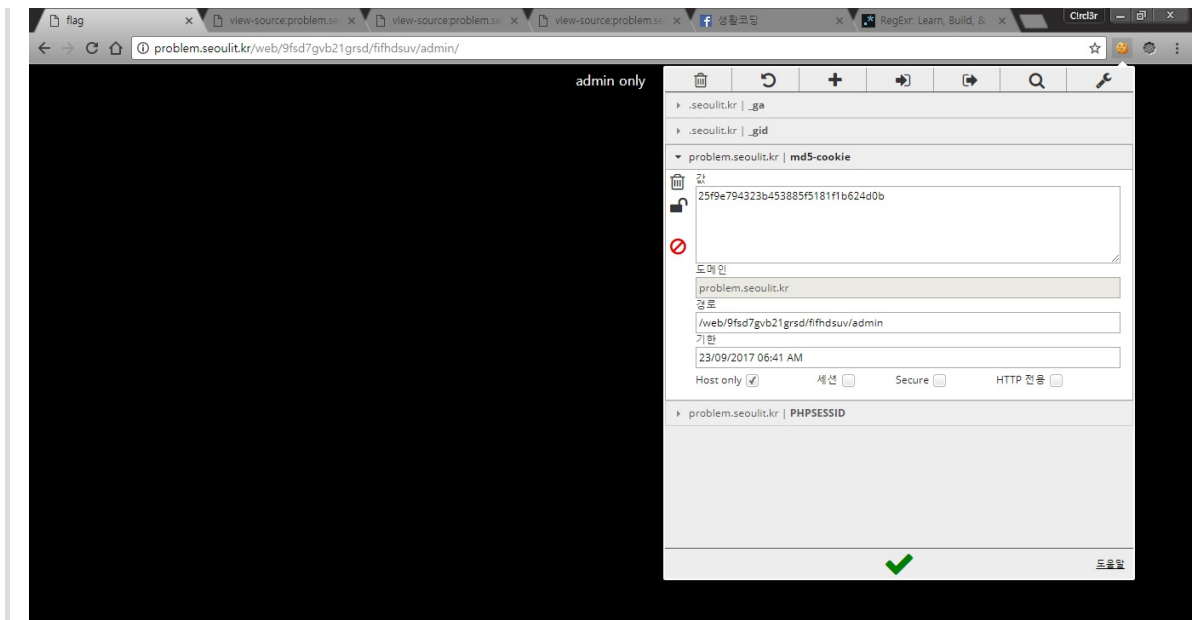
- 일단 웹문제는 웹페이지의 소스코드부터 보는 것이 정석인듯 하다. 우선 보면 robots.txt라고 써져있다. 들어가보자



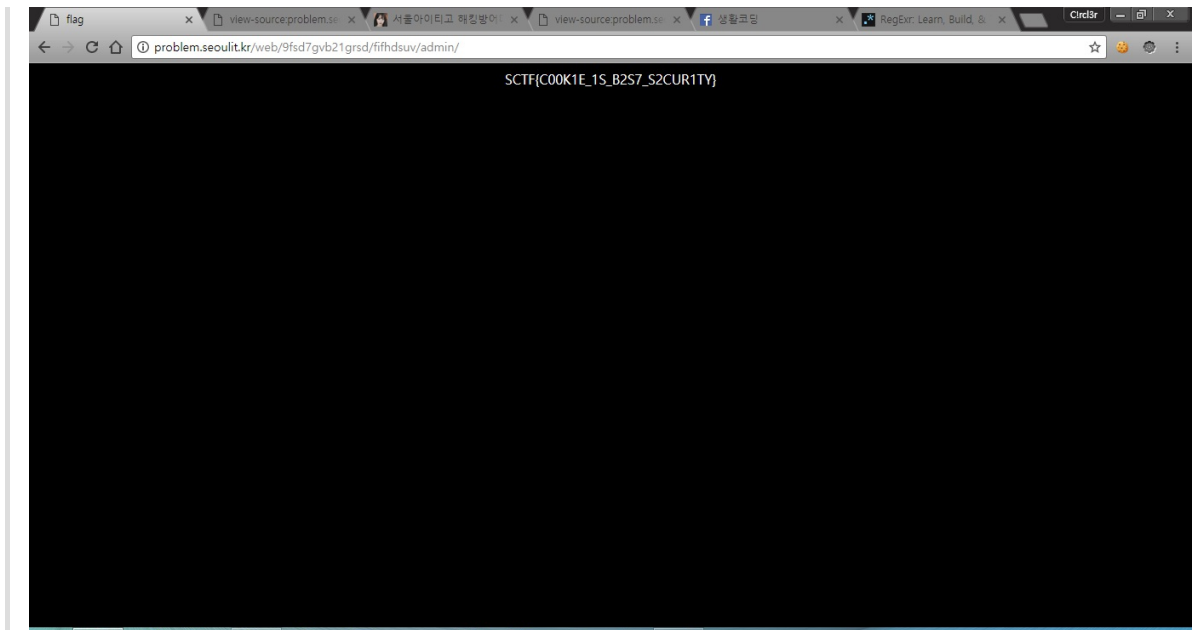
- admin 폴더의 크롤링을 막는다.



- admin 폴더에 들어가서 쿠키를 확인해 보면 내 아이디인 123456789 를 md5로 해싱한 값을 쿠키로 저장했다.

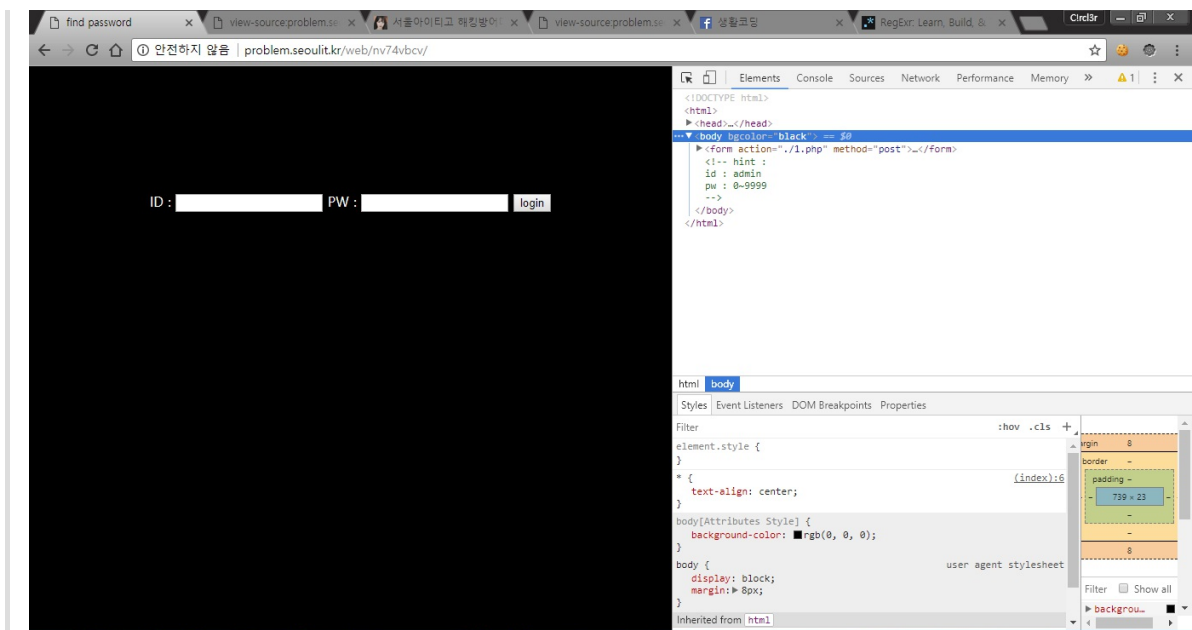


- 그러면 admin 문자열을 md5로 해싱한 값을 쿠키에다가 넣어주면 풀린다.



## 10. find.

- 로그인창이 있고 힌트로 id=admin, password=0~9999 라는 힌트가 주어졌다.



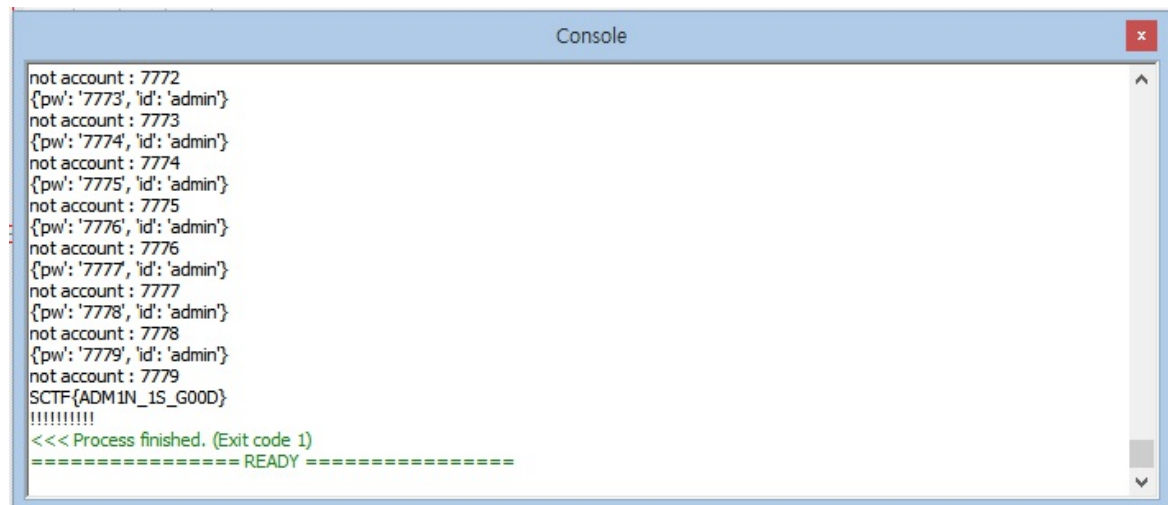
- 뭔가 브루트포싱을 해야 할 것 같다.

```

import requests
import sys
URL = "http://problem.seoulit.kr/web/nv74vbcv/1.php"
for a in range(0,10):
 for s in range(0,10):
 for d in range(0,10):
 for f in range(0,10):
 data = {'id': 'admin', 'pw': '{0}{1}{2}{3}'.format(a,s,d,f)}
 res = requests.post(URL, data)
 res=res.text
 #result = res.read().decode("utf-8")
 if res.find("not account") != -1:
 print(data)
 print(res+" : {0}{1}{2}{3}".format(a,s,d,f))
 else:
 print(res)
 print("!!!!!!!!!!!!")
 sys.exit(1)

```

- 돌리면 나온다.



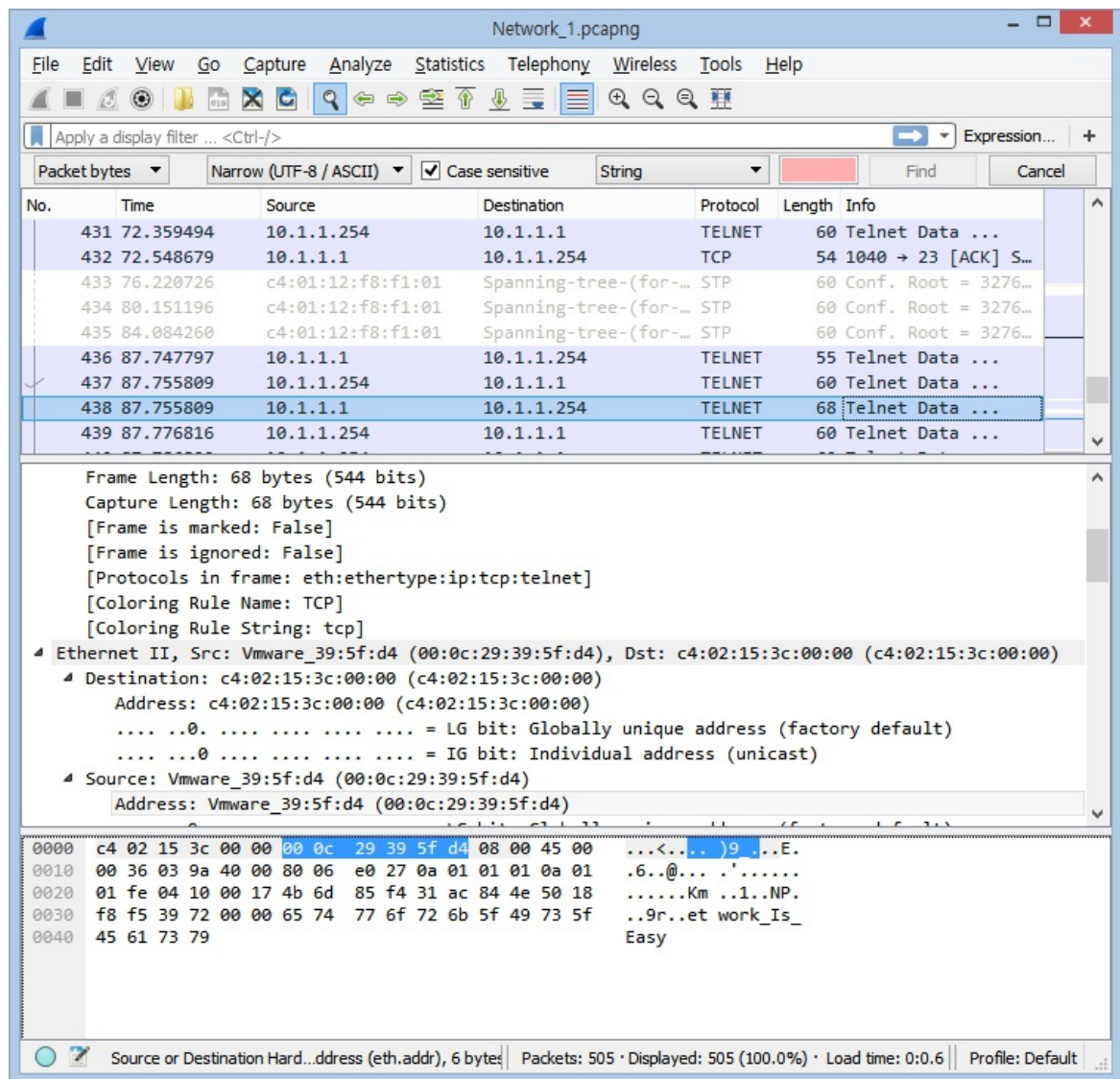
```

Console
not account : 7772
{pw: '7773', id: 'admin'}
not account : 7773
{pw: '7774', id: 'admin'}
not account : 7774
{pw: '7775', id: 'admin'}
not account : 7775
{pw: '7776', id: 'admin'}
not account : 7776
{pw: '7777', id: 'admin'}
not account : 7777
{pw: '7778', id: 'admin'}
not account : 7778
{pw: '7779', id: 'admin'}
not account : 7779
SCTF{ADMIN_IS_GOOD}
!!!!!!!!!!!!
<<< Process finished. (Exit code 1)
===== READY =====

```

## 15. server

- 패킷파일을 하나 주는데 처음에 HEX 에디터로 열어보았다. 그러나 찾기 힘들어서 wireshark로 다시 열었다.

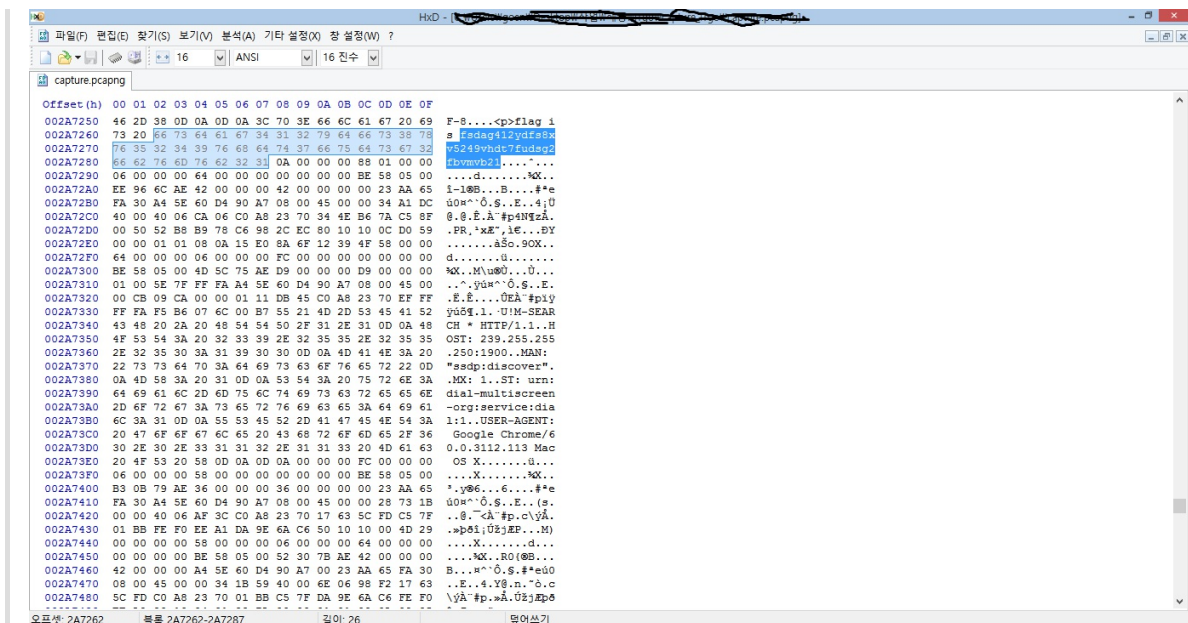


- 보다보면 "..etw ork\_Is\_Easy"라는 문자열이 있는데 약간의 계심으로 netw ork\_Is\_Easy로 인증했더니 풀렸다.

## 16 . capture

- 또 다시 패킷파일을 하나 준다.

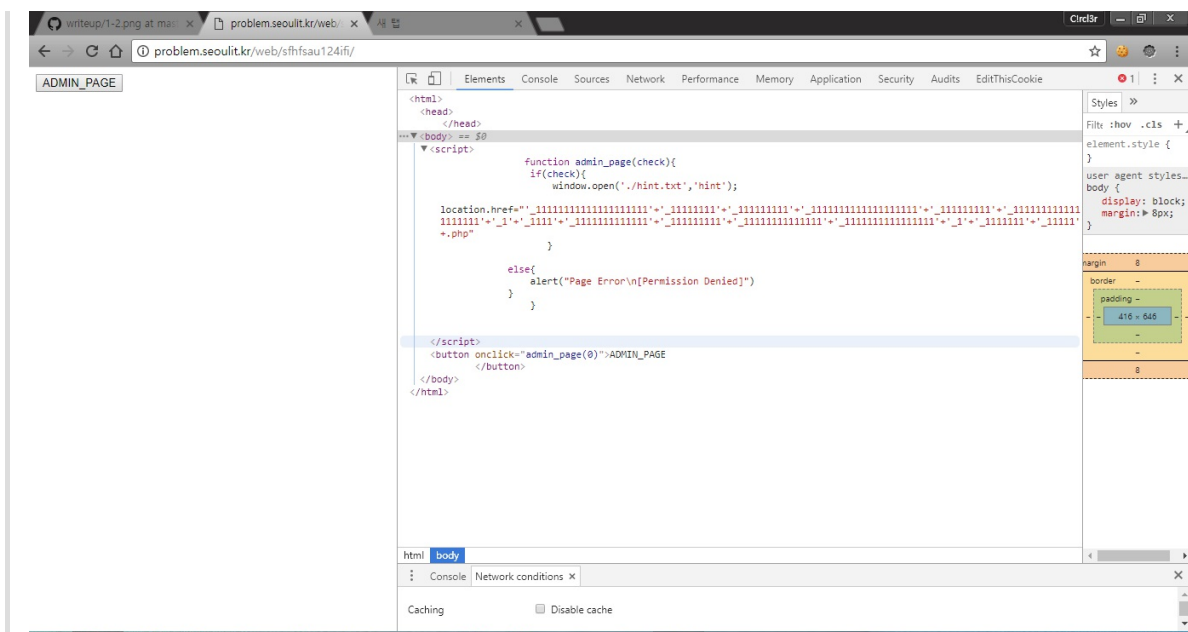




- 열어서 string 검색으로 flag 를 검색했더니 플래그가 나온다.

## 17. unknown

- 페이지에 들어가서 소스코드를 보면 뭔가 이상한 페이지로 리다이렉트시키는데 정상적인 이름이 아니다.



- hint.txt가 있다고 해서 들어갔더니 이상한 문자열이 있다.



writeup/1-2.png at ma... x battienet x problem.seoulit.kr/web... x 새 탭 x Circle

← → ↻ ⚙️ 🔒 안전하지 않음 | problem.seoulit.kr/web/sfhfsau124ifi/thisisadminpage.php ☆ 🛡️ ⋮

[downloadp](#)

[hint]  
id : admin  
password : \*\*\*\*

id   
password

SCTF{LOST\_MY\_PASSWORD\_T\_T}

입력창에다가 admin 과 아무문자열이나 입력하면 플래그가 나온다.