병욱의 시련 답안

1층 : 탑의 문 난이도 : ★☆☆

풀이1

페이지 소스 보기 -> Base64로 인코딩 되어 있는 문자 발견



디코딩 후 anonymity (익명)라는 답 획득

극복

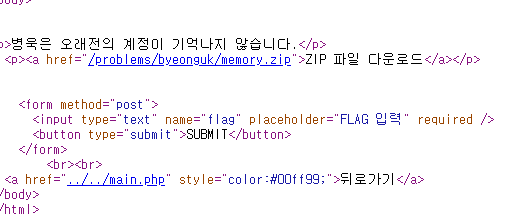
풀이2

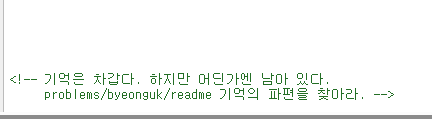
유추해서 풀어낸다. (이름= 가짜, 진실은 이름이 없다 ->익명 ->

극복

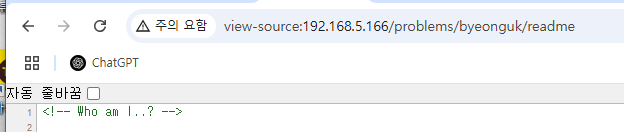
2층 : 차가운 기억 난이도 : ★☆☆

풀이



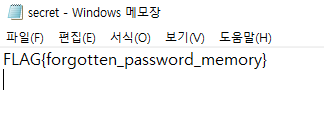


페이지 소스에서 경로를 확인



url을 해당 경로로 수정 -> 소스에서 ‘나는 누구지?’라는 힌트를 받음 ->병욱의 시련 ->병욱? ->byeonguk 추론

압축 해제



FLAG 획득

시련 극복

3층 : 균열 시련 난이도 : ★☆☆

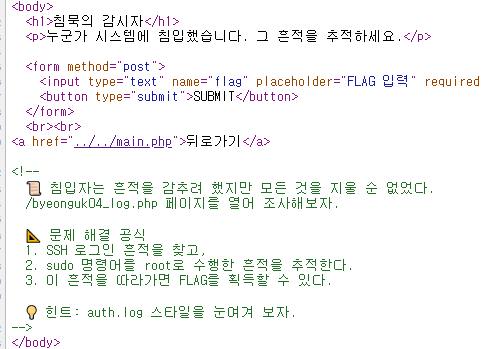


소스에서 수상한 비밀 경로 발견  

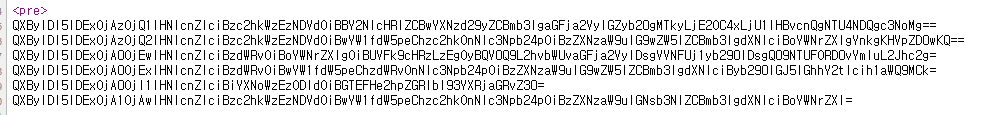

플래그 획득

4층 : 포트의 속삭임 난이도 : ★☆☆

풀이

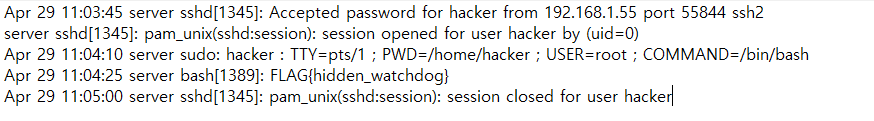


소스 확인



View-source:http://192.168.5.166/problems/byeonguk/byeonguk04\_log.php

로 입장(인코딩 된 글들 확인)

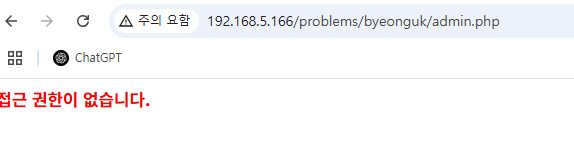


한 줄 씩 디코딩 해본 결과 플래그를 획득 하게 된다.

극복

5층 : 어긋난 질서 난이도 : ★★☆



소스 확인 -> set\_user.php로 접속했으나 기존 페이지로 이동

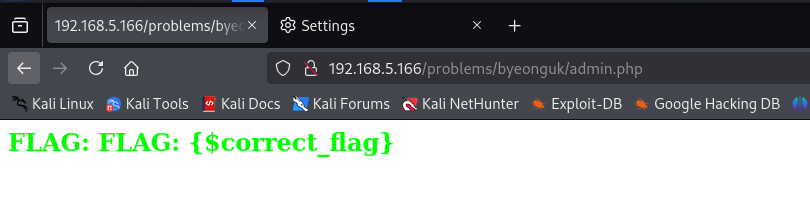
Admin이 되어야 한다는 말을 통해 admin.php를 유추 해서 접속->권한 없음





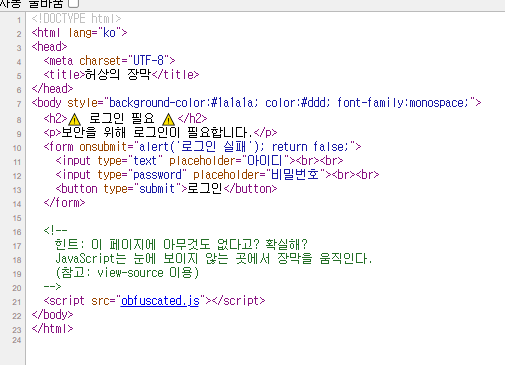
Zaproxy를 통해 브레이크를 잡는다.



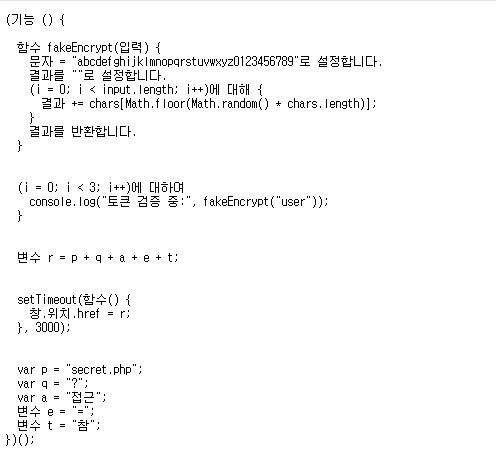
힌트 기반으로 쿠키 값을 수정 후 넘긴다.  


극복

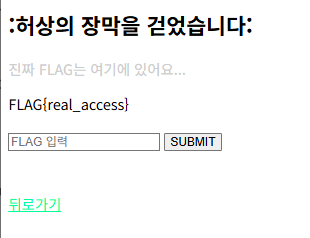
6층 : 로그의 미로 난이도 : ★★☆



페이지 소스를 봤을 때 해당 페이지의 로그인 화면은 아무 의미 없다는것을 확인 할 수 있고 obfuscated.js로 들어가 본다.

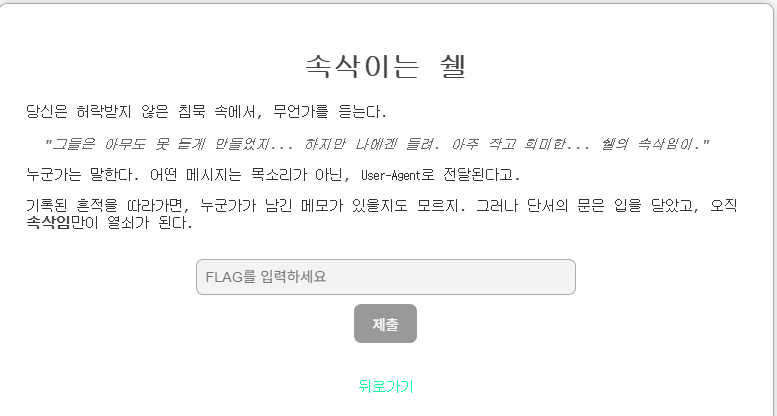


모든 것을 더했을 때 r=secret.php?access=true 이라는 것을 알 수 있다. url에서 해당 페이지로 접속

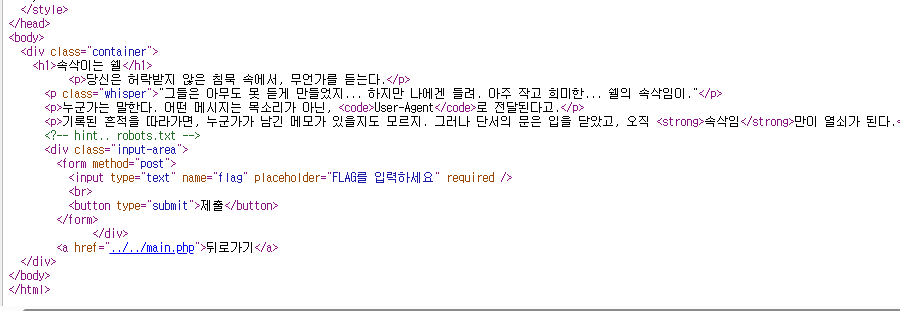


극복

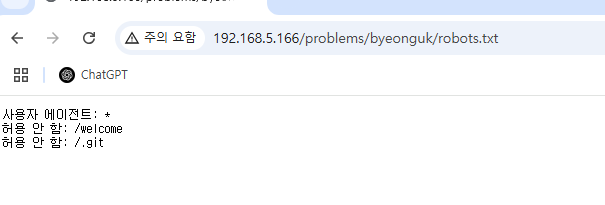
7층 : 속삭이는 쉘 난이도 : ★★☆



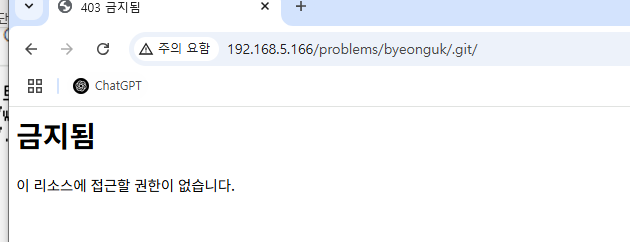
페이지 접속



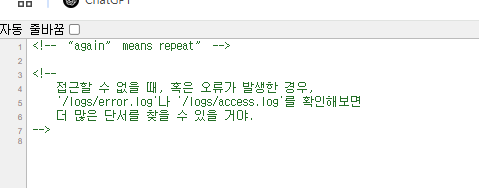
Robots.txt로 들어가 보라는 힌트 확인



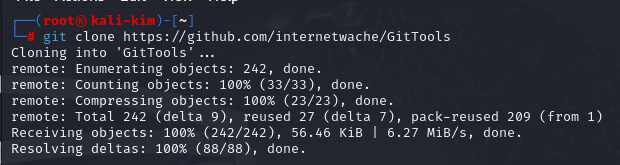
새로운 경로들을 발견



금지 됨



welcome으로 접속 했을 때 해당 디렉토리로 접속 하라고 했지만 찾을 수 없었음



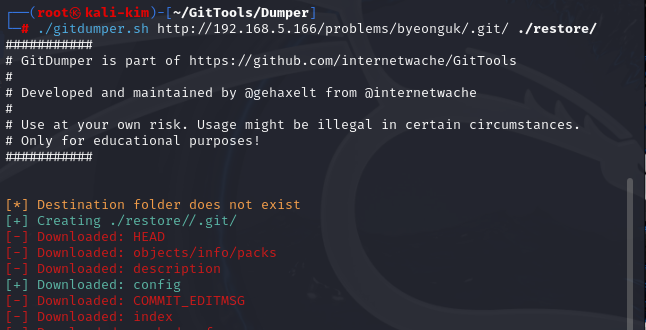
git-dumper로 .git 폴더 복구하기

.git/ 폴더는 웹사이트의 모든 역사(코드 변경 내용)를 담고 있음  
 원래 막아놔야 되는데, 실수로 열려 있으면 그걸 **복구**할 수 있음.

일단 kali에서 깃툴즈(GitTools)라는 **도구를 내 컴퓨터로 복사해오기.**



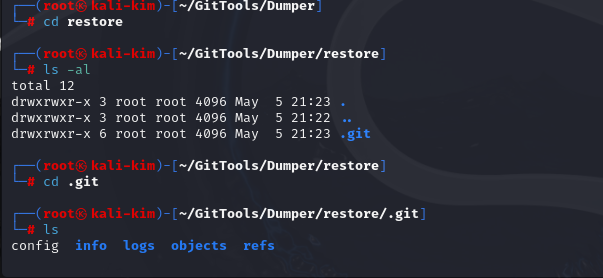
cd GitTools/Dumper 명령어로 Dumper에 입장

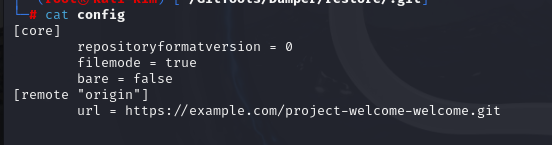


./gitdumper.sh http://<서버주소>/problems/byeonguk/.git/ ./restore/

.git/ 폴더 내용을 **웹에서 긁어서** restore/ 폴더에 **복구해줘!라는 명령어 시전.**  
:파일_폴더: 복구가 끝나면, restore 폴더 안에 진짜 웹페이지 소스코드가 나옴.  
 그걸 열어보면서 힌트도 찾고, 보안 실수도 캐낼 수 있음.

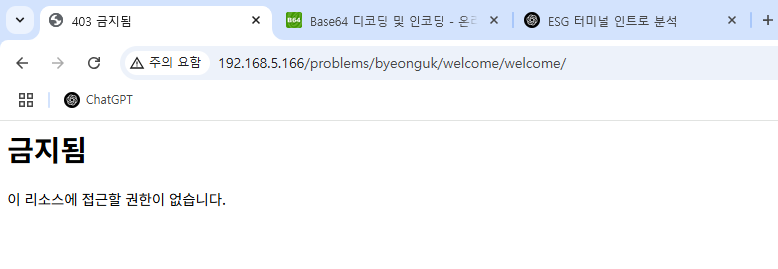
config파일이 다운 된 것을 확인 할 수 있음.



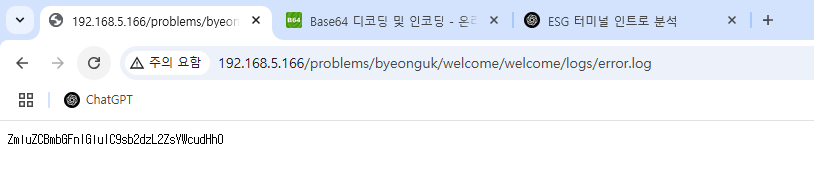
복원 된 파일을 발견  


**url = https://example.com/project-welcome-welcome.git 이 의미하는 것**

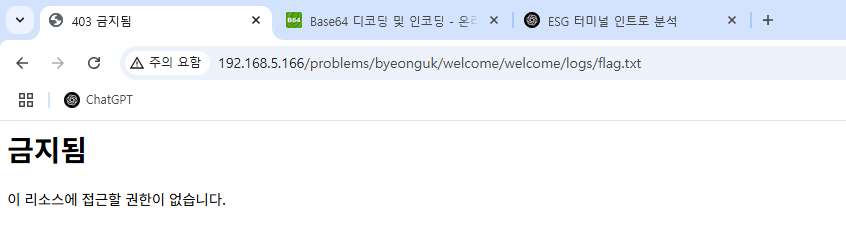
**"welcome-welcome.git"이라는 프로젝트 이름**  
→ 프로젝트명 또는 디렉토리 구조가 **welcome/welcome** 형태일 가능성을 강하게 암시



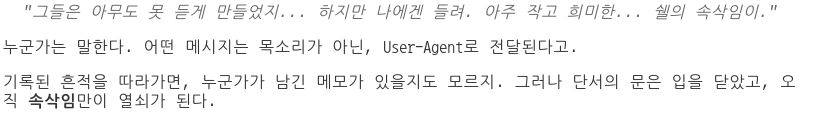
접근은 금지 되었으나 존재 한다는 것을 확인



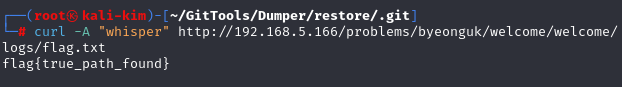
막혔을 때 logs/error.log 또는 logs/access.log로 가보라는 안내를 통해서 이동 ->디코딩 시 find flag in /logs/flag.txt



막혀있음



일반적으로 접속 했을 때는 닿지 못하지만 User-Agent가 속삭임(whisper)이라면 접속 할 수 있을 것 같음.



curl -A "whisper" http://<서버주소>//problems/byeonguk/welcome/welcome/logs/flag.txt

나는 whisper라는 브라우저이고 해당 주소의 페이지를 보여줘

극복

8층 조작된 시련 — Trial of Tamper

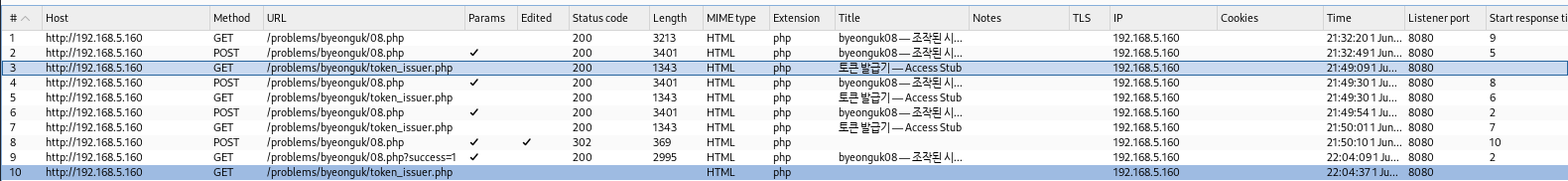
난이도 : ★★★



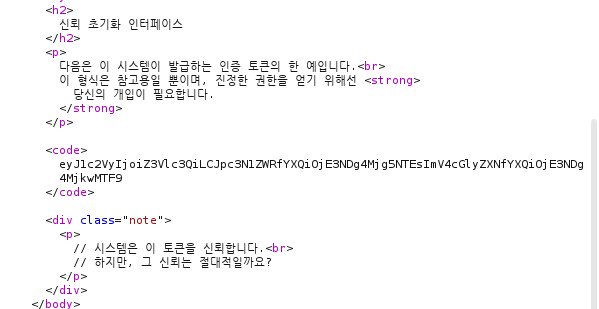
시련에 접속 했을 때 소스보기, F12에서의 소스코드 보기가 막힌 상황



토큰 입력 란에 토큰을 잡아서 봤을 때도 “token=(내가 넣은 값)” 외에는 특별 한 것이 없는 상황



http history로 넘어 가보면 특별한 URL 기록이 있는 것을 확인 할 수 있음



Base64로 인코딩 된 code와 개입을 유도하는 힌트를 얻는다.

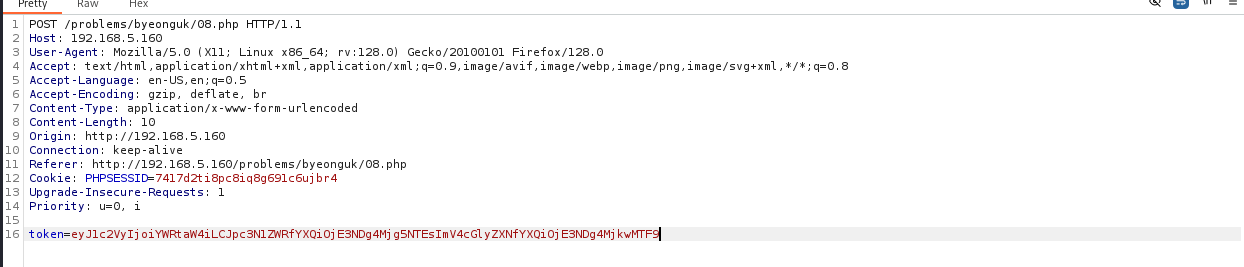


디코딩 결과

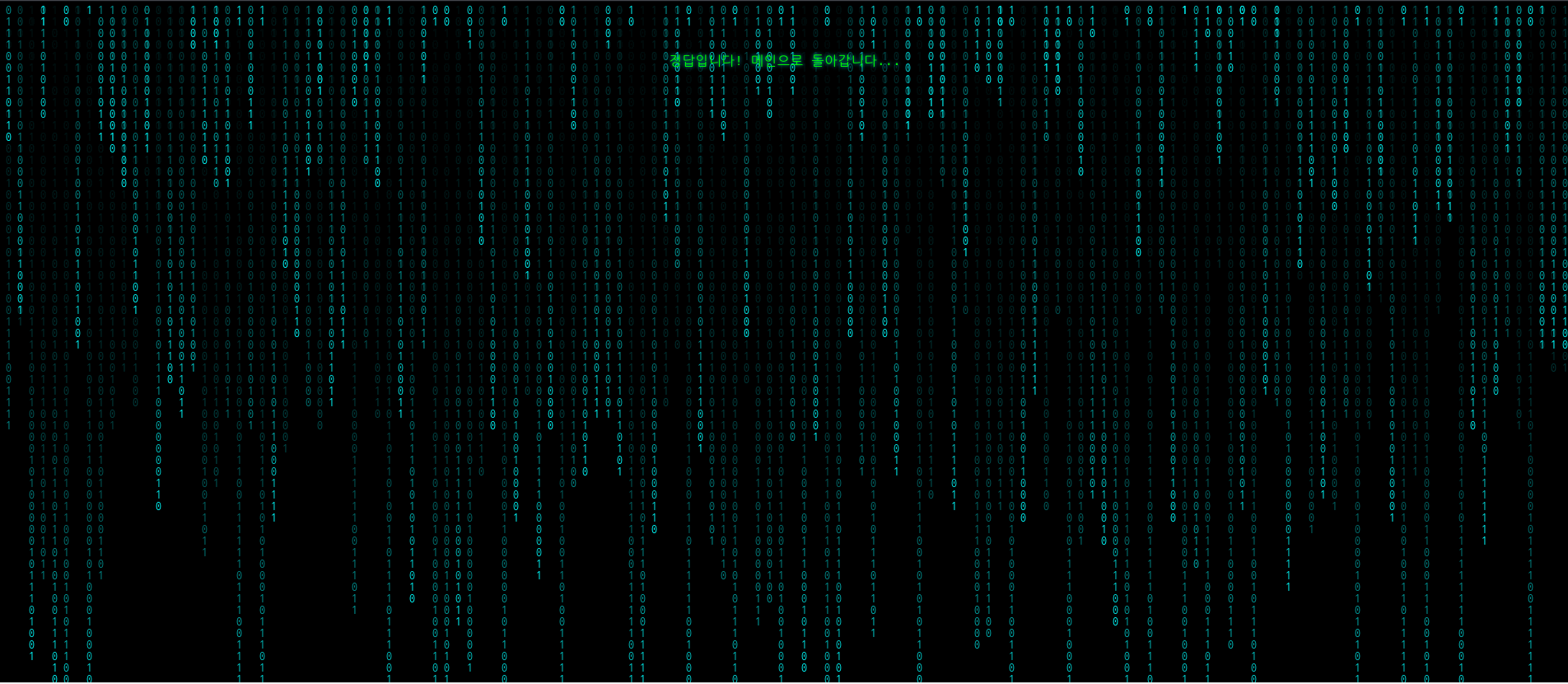
"user": "guest",

"issued\_at": 1717405000,

"expires\_at": 1717405600 토큰 인증 정보라는 생각을 할 수 있다.



게다가 맨 앞에서 admin이냐 그렇게 만들 수 있는 존재냐 라는 언급과 진정한 권한을 얻기 위해서 개입하라는 말들을 통해서 user를admin으로 고쳐 보자 라는 생각을 할 수 있게 된다. 그리고 처음 코드를 받았을 때 base64로 인코딩 되어있었으므로 대칭을 위해서 admin으로 고친 값을 base64로 다시 인코딩 해 진행.

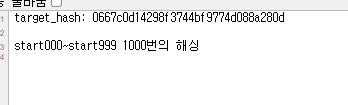


극복 (토큰 입력란에 직접 작성해도 됨)

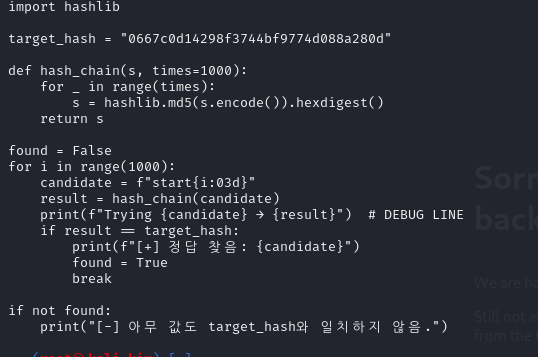
9층 : 가면을 벗겨라 난이도 : ★★★



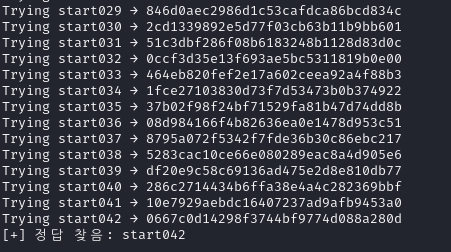
Assets/hash\_chain.txt로 이동



start000~start999 중 한 숫자와 일치한다.



kali에서 target\_hash의 start000~start999까지 1000번 해싱 한 값과 비교 해서 일치하는 값을 찾는 코드 작성

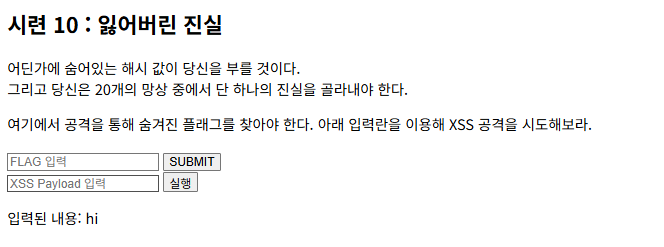


코드 작성 완료 후 python3 파일이름.py 명령어를 통해서

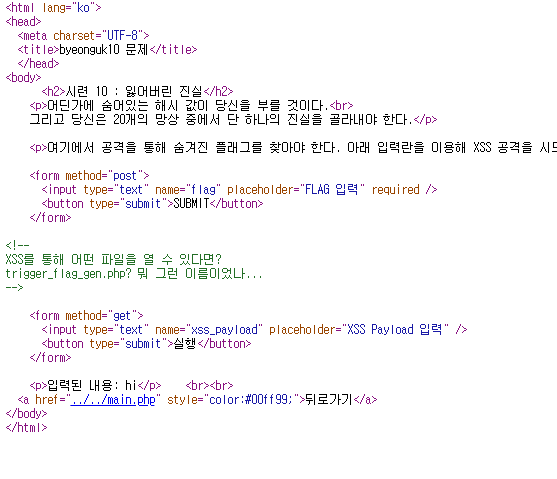
일치 하는 값 발견

극복

10층 : 본질의 시험 난이도 : ★★★



입력한 글이 출력 된다.



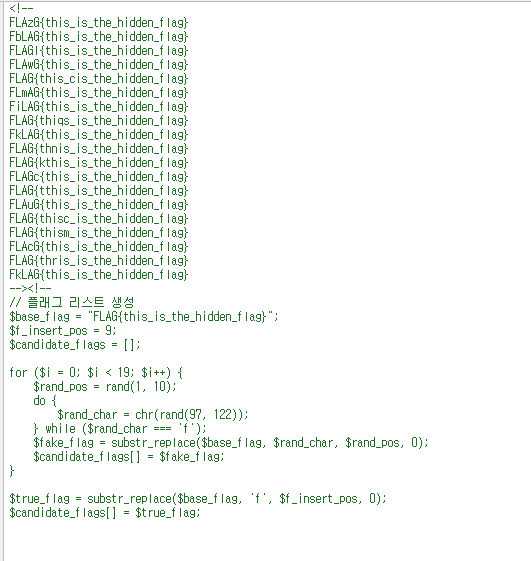
XSS를 통해서 trigger\_flag\_gen.php 페이지로 넘어가야 한다는 걸 알게 됨

<script>window.location.href = ‘/problems/byeonguk/trigger\_flag\_gen.php’;</script> 작성 후 실행 -> 해당 페이지로 넘어감

## window.location.href

## 구성요소

* window: 현재 브라우저 창(또는 탭)을 의미하는 전역 객체입니다.
* location: 현재 문서의 URL 및 이동 관련 속성을 가진 객체입니다.
* href: 현재 URL을 가리키는 문자열 속성입니다.



작성 되어 있는 코드를 읽어보면 FLAG{this\_is\_the\_hidden\_flag} 에다가 위에 제시 되어있는 19개의 가짜 FLAG들임이 확인 되었고

진짜는 주석으로 출력 되지 않은 FLAG{this\_is\_the\_hidden\_flag}에서 9번째 글자에 f를 추가한 FLAG{thisf\_is\_the\_hidden\_flag}가 진짜 flag 임을 확인 할 수 있음.

극복