

攻防世界 Reverse高手进阶区 2分题 parallel-comparator-200

原创

思源湖的鱼 于 2020-12-07 14:01:51 发布 95 收藏

分类专栏: [ctf](#) 文章标签: [reverse](#) [ctf](#) [攻防世界](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 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/weixin_44604541/article/details/110816945

版权

CTF

[ctf 专栏收录该内容](#)

200 篇文章 23 订阅

订阅专栏

前言

继续ctf的旅程

攻防世界Reverse高手进阶区的2分题

本篇是parallel-comparator-200的writeup

发现攻防世界的题目分数是动态的

就仅以做题时的分数为准了

解题过程

得到一段c

```
#include <stdlib.h>
#include <stdio.h>
#include <pthread.h>

#define FLAG_LEN 20

void * checking(void *arg) {
    char *result = malloc(sizeof(char));
    char *argument = (char *)arg;
    *result = (argument[0]+argument[1]) ^ argument[2];
    return result;
}

int highly_optimized_parallel_comparision(char *user_string)
{
    int initialization_number;
    int i;
```

```

char generated_string[FLAG_LEN + 1];
generated_string[FLAG_LEN] = '\0';

while ((initialization_number = random()) >= 64);

int first_letter;
first_letter = (initialization_number % 26) + 97;

pthread_t thread[FLAG_LEN];
char differences[FLAG_LEN] = {0, 9, -9, -1, 13, -13, -4, -11, -9, -1, -7, 6, -13, 13, 3, 9, -13, -11, 6, -7}
;

char *arguments[20];
for (i = 0; i < FLAG_LEN; i++) {
    arguments[i] = (char *)malloc(3*sizeof(char));
    arguments[i][0] = first_letter;
    arguments[i][1] = differences[i];
    arguments[i][2] = user_string[i];

    pthread_create((pthread_t*)(thread+i), NULL, checking, arguments[i]);
}

void *result;
int just_a_string[FLAG_LEN] = {115, 116, 114, 97, 110, 103, 101, 95, 115, 116, 114, 105, 110, 103, 95, 105,
116, 95, 105, 115};
for (i = 0; i < FLAG_LEN; i++) {
    pthread_join(*(thread+i), &result);
    generated_string[i] = *(char *)result + just_a_string[i];
    free(result);
    free(arguments[i]);
}

int is_ok = 1;
for (i = 0; i < FLAG_LEN; i++) {
    if (generated_string[i] != just_a_string[i])
        return 0;
}

return 1;
}

int main()
{
    char *user_string = (char *)calloc(FLAG_LEN+1, sizeof(char));
    fgets(user_string, FLAG_LEN+1, stdin);
    int is_ok = highly_optimized_parallel_comparsion(user_string);
    if (is_ok)
        printf("You win!\n");
    else
        printf("Wrong!\n");
    return 0;
}

```

看了看

按这意思写个脚本就是了

```

differences = [0, 9, -9, -1, 13, -13, -4, -11, -9, -1, -7, 6, -13, 13, 3, 9, -13, -11, 6, -7]
first_letter = 108
print (''.join([chr(first_letter+i) for i in differences]))

```

```
1 differences = [0, 9, -9, -1, 13, -13, -4, -11, -9, -1, -7, 6, -13, 13, 3, 9, -13, -11,  
2 first_letter = 108  
3 print (''.join([chr(first_letter+i) for i in differences]))
```

lucky_hacker_you_are

得到flag

结语

简单题