攻防世界逆向入门题之open-source

原创

沐一 · 林 于 2021-08-04 14:29:17 发布 75 收藏

分类专栏： CTF 逆向 文章标签： unctf

本文链接：https://blog.csdn.net/xiao__1bai/article/details/119383000

CTF 同时被 2 个专栏收录

167 篇文章 6 订阅

订阅专栏

逆向

95 篇文章 6 订阅

订阅专栏

攻防世界逆向入门题之open-source

继续开启全栈梦想之逆向之旅~

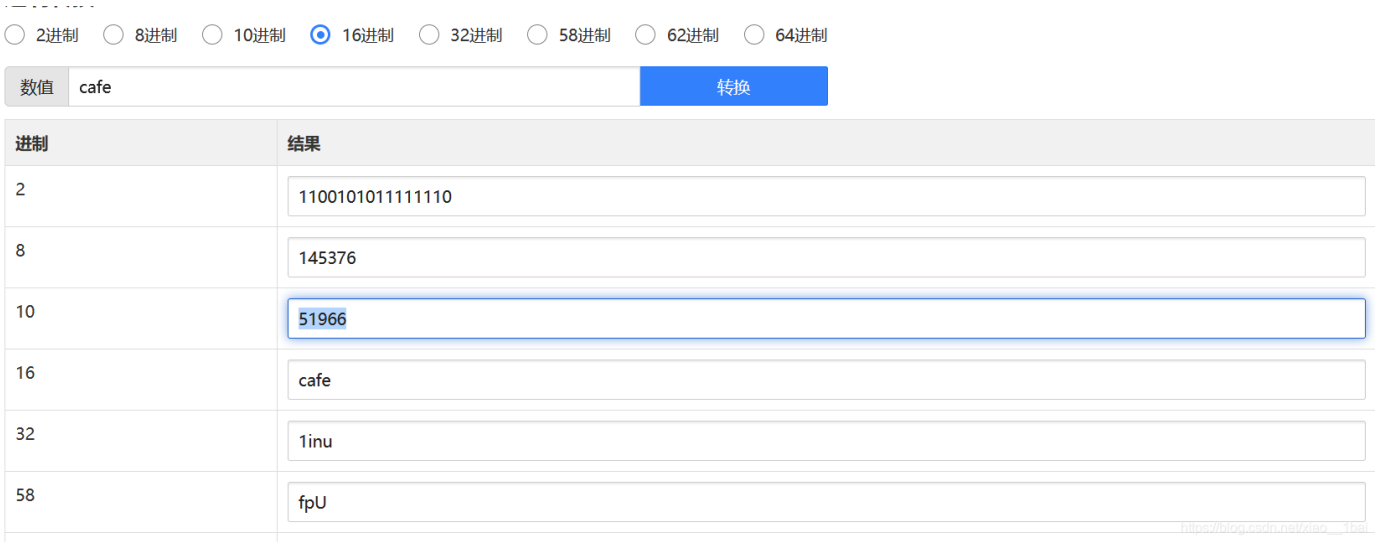
这题是攻防世界逆向入门题的open-source

下载附件得到源码：

```c
#include <stdio.h>
#include <string.h>

int main(int argc, char *argv[]) {//外部调用输入参数
    if (argc != 4) {//输入三个参数，因为第一个是程序自己的名称
     printf("what?\n");
     exit(1);
    }

    unsigned int first = atoi(argv[1]);
    if (first != 0xcafe) {//第一个参数的十六进制为0xcafe
     printf("you are wrong, sorry.\n");
     exit(2);
    }

    unsigned int second = atoi(argv[2]);
    if (second % 5 == 3 || second % 17 != 8) {//第二个参数满足条件我口算有42，余数是不足才补的数，不是整除后剩的数。
     printf("ha, you won't get it!\n");
     exit(3);
    }

    if (strcmp("h4cky0u", argv[3])) {//第三个参数直接就是h4cky0u
     printf("so close, dude!\n");
     exit(4);
    }

    printf("Brr wrrr grr\n");

    unsigned int hash = first * 31337 + (second % 17) * 11 + strlen(argv[3]) - 1615810207;//这里的结果hash与前面输入参数有关，鄙人不才，曾一度想修改源码不输入参数直接输出这句话，当然，没有参数的这句话就会报错。

    printf("Get your key: ");
    printf("%x\n", hash);
    return 0;
}
```

一开始第二个条件停了会，毕竟做题经验太少了，atoi返回的是字符串的整形，0xcafe是十六进制，整形和十六进制比较C语言内部会进行进制转换，一开始我用了十六进制转文本：

现在想想都羞愧~
应该用进制转换工具才对：



所以到此所有参数都解出来了，第一个是51966，第二个是42，第三个是h4cky0u.
在kali虚拟机中编译，命令行接受参数执行即可：

```
gcc 1.c
./1.c 51966 42 h4cky0u
```

后来看别人做法还发现了其他解法，第一个是直接修改源码，其实也对，源码在手当然是充分利用源码的优势才对，直接把hash输出语句替换成

```
unsigned int hash = 0xcafe * 31337 + (second % 17) * 11 + strlen(argv[3]) - 1615810207;
```

即可，反正C语言内部会自己转换，记得把第二个0xcafe处的判断语句用/**/注释掉即可。