# 攻防世界 逆向 open-source

原创

与日平肩以头抢地 于 2020-01-29 10:22:05 发布 2534 收藏 1

文章标签： python c语言



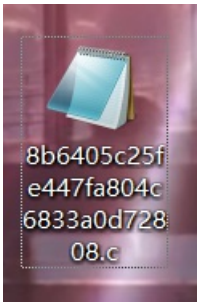
本文链接：https://blog.csdn.net/fool_best/article/details/104104789



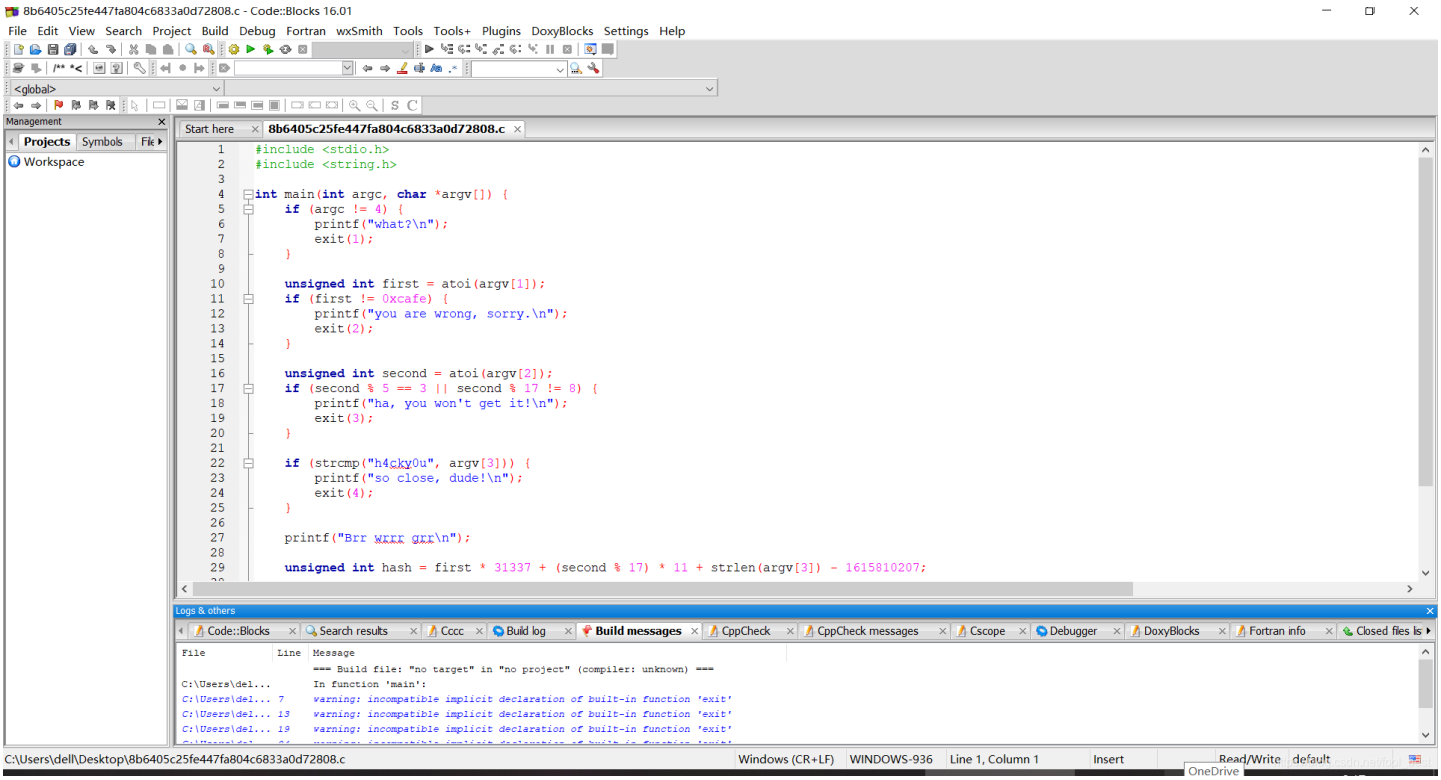
## 攻防世界 逆向 open-source

原创


首先当然是下载附件




是一个.c类型的文件，打开codeblock将此文件拖进去查看代码

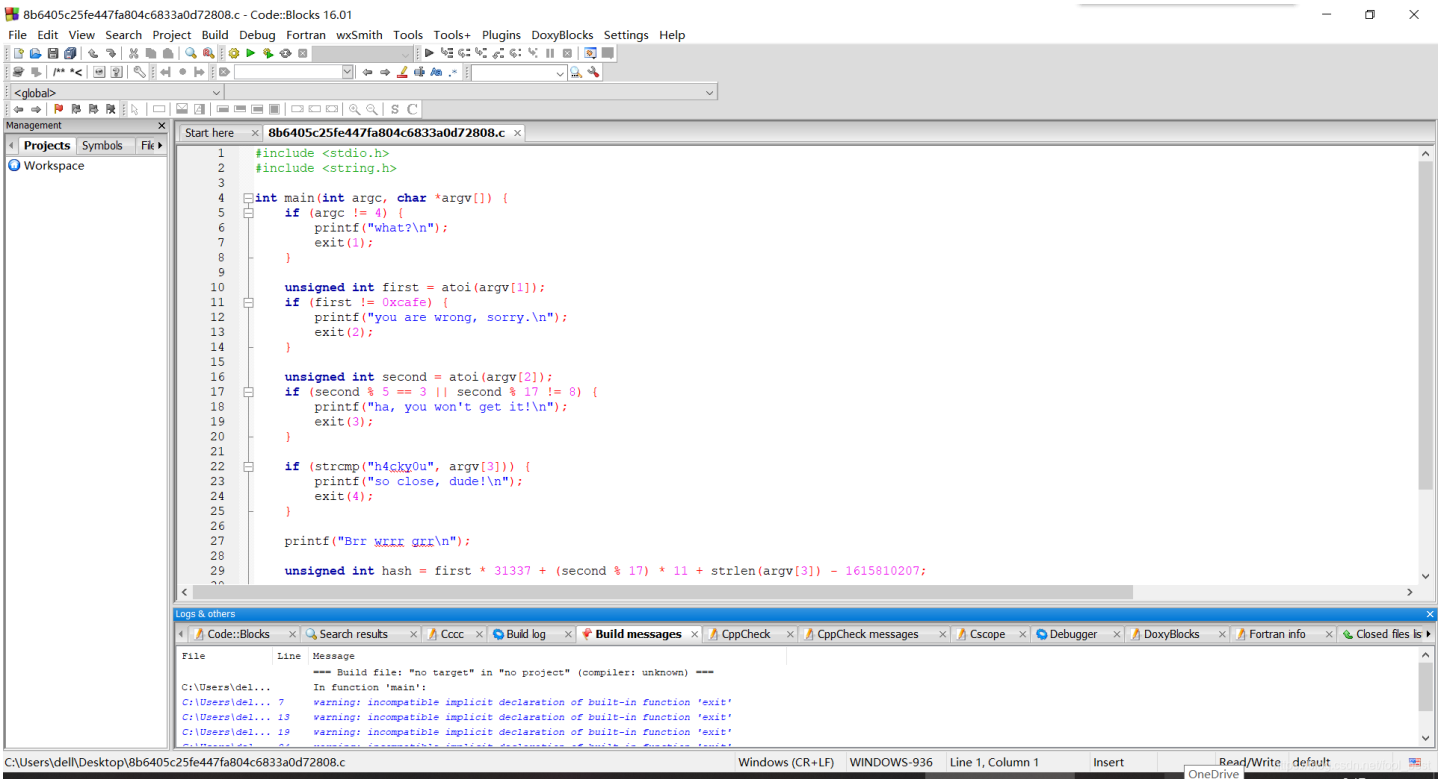



代码不多，运行之后程序打印“what?”，这。。。并没有什么大用处
通过观察代码，可以发现最后一部分才是解题的关键。

```
    exit(4);
}

printf("Brr wrrr grr\n");

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

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

找到目标了，就是要求hash，这就好办多了。
现在需要求到first，second和strlen(argv[3])的值。

看向上面的代码中有提到那三个变量。



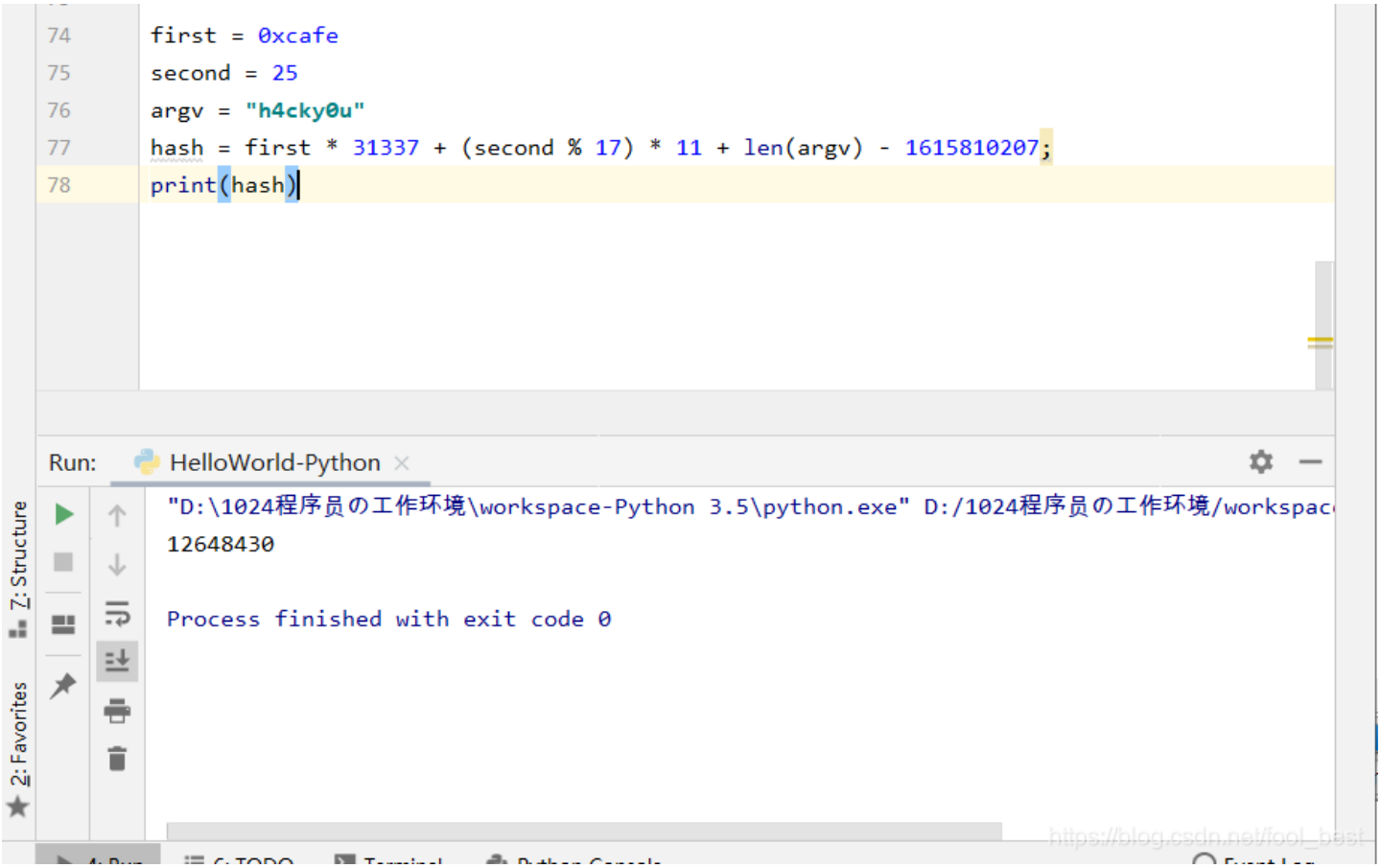

first求法：
源代码如下
unsigned int first = atoi(argv[1]);
if (first != 0xcafe) {
printf(“you are wrong, sorry.\n”);
exit(2);
}
通过if的条件可以知道first就是0xcafe，一个16进制的数字，也就是first的值。

second求法：
源代码如下
unsigned int second = atoi(argv[2]);
if (second % 5 == 3 || second % 17 != 8) {
printf(“ha, you won’t get it!\n”);
exit(3);
}
通过if的条件可以知道，(second % 5 == 3 || second % 17 != 8)应该为false，也就是说(second % 5 == 3)和 (second % 17 != 8)两者都是错的，也就可以得到second%7==8成立，hash中的一部分second%17也是求出来了，就是8，此处暂时将second取一个正确的值为25。

strlen(argv[3])求法：
源代码如下
if (strcmp(“h4cky0u”, argv[3])) {
printf(“so close, dude!\n”);
exit(4);
}
直接可以知道argv[3]==“h4cky0u”，strlen(argv[3])=7。

```
        exit(4);
    }

    printf("Brr wrrr grr\n");

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

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

计算过程中可以有比较大的数字出现，使用C语言可能导致溢出得到错误答案，我们在这里使用python脚本进行运算。




得到结果为12648430，但是这不是flag
接着看下面的源代码

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

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

%x，要求以16进制的方式输出。转为16进制，在python中使用 print(hex(hash))，得到结果0xc0ffee。

（ps：结果中的0xc0ffee的0x说明这个数是16进制的数字，并不是数字部分）