题目描述

checkin challenge

题目考点

基本的逆向能力、脱壳、`dfs`算法

题目writeup

1、对程序脱壳。

```
upx -d  maze.exe -o maze_decompress.exe
```

2、使用IDA进行分析。

地址`00404020`处是一个10x10大小的迷宫，将数据复制出来，编写dfs脚本，解出答案。

```
import numpy as np
maze = [1,   0,   0,   0,   0,   1,   1,   0,   0,   0,
        1,   1,   0,   0,   0,   1,   1,   1,   0,   0,
        0,   1,   0,   0,   0,   1,   0,   1,   0,   0,
        0,   1,   1,   1,   1,   1,   0,   1,   0,   0,
        0,   0,   0,   0,   1,   1,   0,   1,   0,   0,
        0,   0,   0,   0,   0,   0,   0,   1,   0,   0,
        0,   0,   0,   0,   0,   0,   0,   1,   1,   0,
        0,   0,   0,   0,   0,   0,   0,   0,   1,   0,
        0,   0,   0,   0,   0,   0,   0,   0,   1,   1,
        0,   0,   0,   0,   0,   0,   0,   0,   0,   1,]

book = np.zeros((10, 10), np.uint8)
nxt = [(0, 1), (0, -1), (1, 0), (-1, 0)]
flag = []

def dfs(curx, cury):
    if curx == 9 and cury == 9:
        print("".join(flag))
        exit(0)
    for i in range(4):
        new_x = curx + nxt[i][0]
        new_y = cury + nxt[i][1]
        if 0 <= new_x < 10 and 0 <= new_y < 10 and book[new_x][new_y] == 0 and
maze[new_x*10+new_y] == 1:
```

```
            book[new_x][new_y] = 1
            flag.append(str(i))
            dfs(new_x, new_y)
            flag.pop()
            book[new_x][new_y] = 0

if __name__ == "__main__":
    dfs(0, 0)
# flag{2022000033002222202202}
```