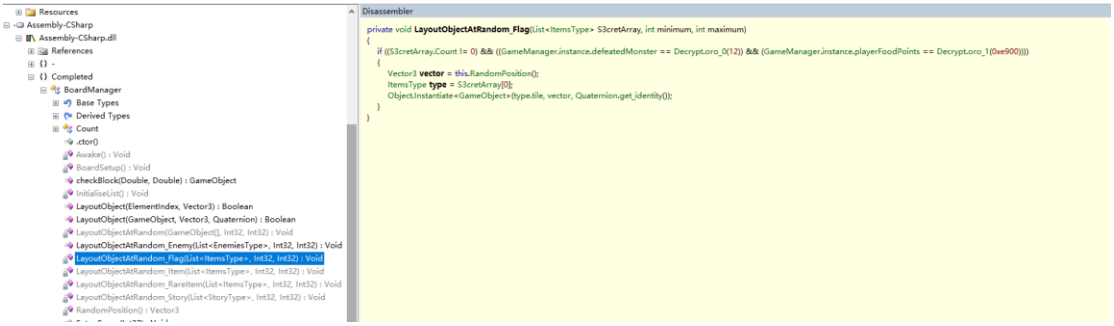
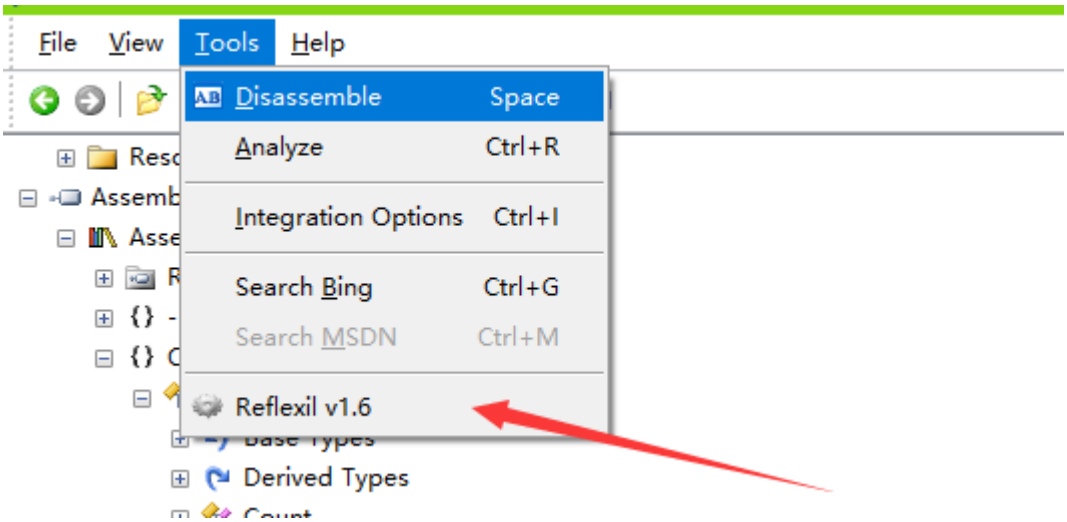
自己看着别人的 writeup 今天终于搞出来了.

官方 writeup

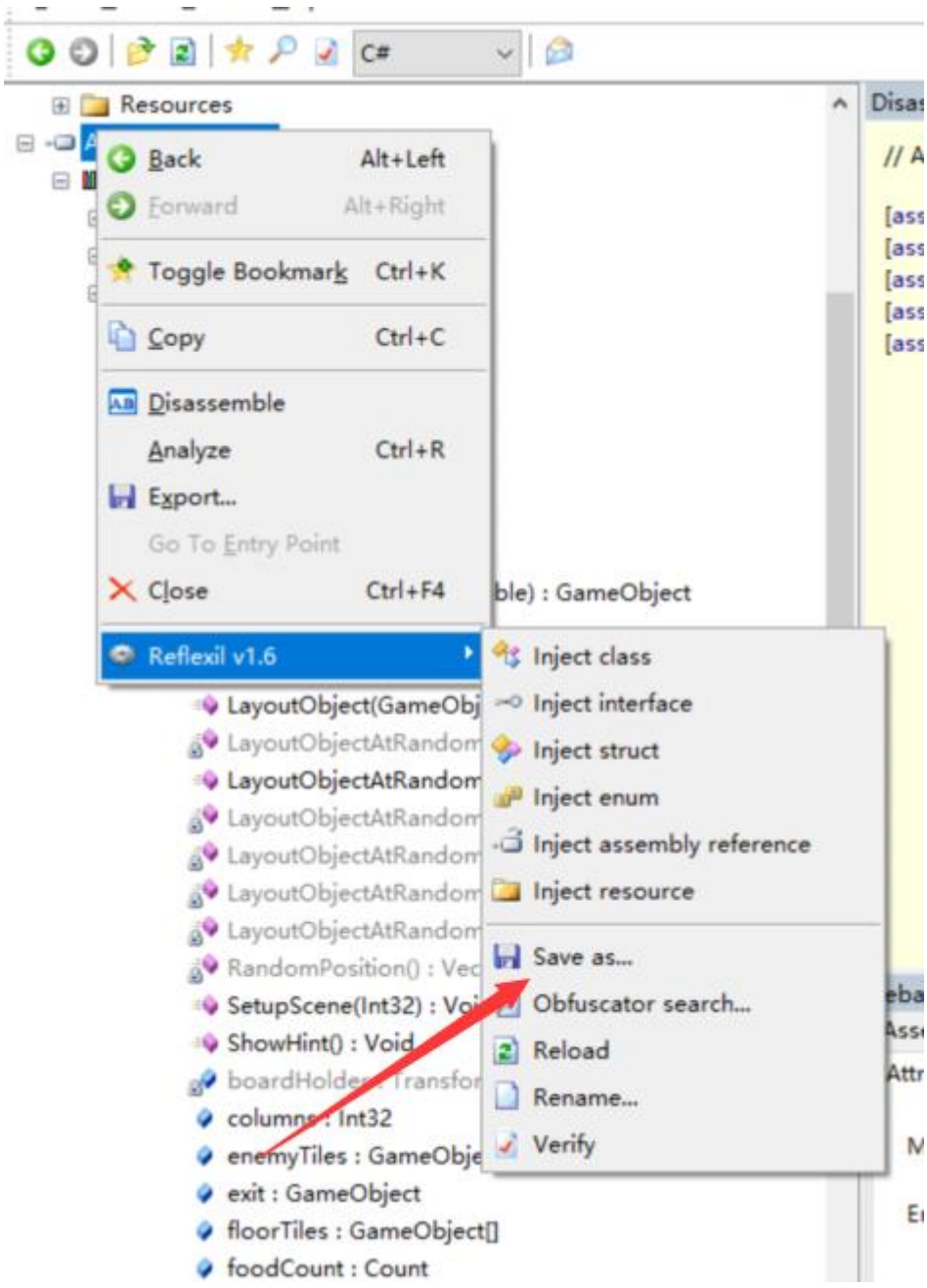
https://www.xctf.org.cn/library/details/2ff21f569a791e21cbd6ce0d4675a9de5ec2373a/

另一种思路:

https://github.com/balsn/ctf_writeup/tree/master/20180526-suctf

1. 玩了很长时间的游戏,才知道 I 键可以看装备,我也是佩服自己.
2. 百度得知 Unity3d 写的 c#脚本都被编译成了 Assembly-CSharp.dll
3. 在\rev_Data\Managed 中发现了 Dotfuscated,后来知道这是一种混淆方式
4. 用 de4dot 得到混淆前的 dll 文件
5. 再用 reflector+reflexil1.6.dll 进行反编译.
6. 
7. 可以清楚地看到这个函数名字已经给了非常明显的提示._Flag
8.
9.
```
private void LayoutObjectAtRandom_Flag(List<ItemsType> S3cretArray, int minimum, int maximum)
{
    if ((S3cretArray.Count != 0) && ((GameManager.instance.defeatedMonster == Decrypt.oro_0(12)) && (GameManager.instance.playerFoodPoints == Decrypt.oro_1(0xe900))))
    {
        Vector3 vector = this.RandomPosition();
        ItemsType type = S3cretArray[0];
        Object.Instantiate<GameObject>(type.tile, vector, Quaternion.get_identity());
    }
}
```
10. 看源码发现,只有两个逻辑判断正确,才会将 S3cretArray[0];的内容显示出来.
11. 

12. 然后用 reflexil 修改指令并保存

13. 

14. 后来发现直接使用去掉混淆的 dll 会出错,只能看懂反混淆后的 dll 去改最初的 dll 了

15.

	Offset	OpCode	Operand
30	116	ldc.i4	6
31	121	stloc.3	
32	122	br.s	-> (4) ldloc.3
33	124	ldsfld	Completed.GameManager Completed.GameManager::instance
34	129	callvirt	System.Int32 Completed.GameManager::get_defeatedMonster()
35	134	ldc.i4.s	12
36	136	call	System.Int32 Decrypt::oro_0(System.Int32)
▶ 37	141	beq	-> (62) ret
38	146	ldc.i4	2
39	151	stloc.3	
40	152	br	-> (4) ldloc.3
41	157	br.s	-> (18) ldc.i4 3
42	159	ret	

16. 修改 il 指令

    Bne.Un 当两个无符号整数值或不可排序的浮点型值不相等时，将控制转移到目标指令。

    Beq 如果两个值相等，则将控制转移到目标指令。

17. 修改完成后,进去会看到一个小红旗

18. 

19. 

20. Base64 解码后得到了一半 flag
21. 第二部分 flag 在游戏开始的动画中提到了一个 spell
22. 在源程序中发现又一个变量为 SPText 在 函数 initGame 中初始化为 false
23. Disassembler

```
private void InitGame()
{
    this.doingSetup = true;
    this.levelImage = GameObject.Find("LevelImage");
    this.levelText = GameObject.Find("LevelText").GetComponent<Text>();
    this.SPText = GameObject.Find("SPELLText").GetComponent<Text>();
    this.levelText.set_text("Day " + this.level);
    this.SPText.set_enabled(false);
    this.levelImage.SetActive(true);
    base.Invoke("HideLevelImage", this.levelStartDelay);
    this.enemies.Clear();
    this.fadinglist.Clear();
    BagSystem.instance.Initialize();
    this.controllerScript.initialize();
    this.boardScript.SetupScene(this.level);
    Debug.Log(this.dMonster);
}
```

24.

33	112	callvirt	System.Void UnityEngine.Behaviour::set_enabled(System.Boolean)
34	117	ldarg.0	
35	118	ldfld	UnityEngine.GameObject Completed.GameManager::levelImage
36	123	ldc.i4.1	
▸ 37	124	callvirt	System.Void UnityEngine.GameObject::SetActive(System.Boolean)
38	129	ldarg.0	
39	130	ldstr	HideLevelImage
40	135	ldarg.0	

其中发现了

Ldc.I4.0 将整数值 0 作为 int32 推送到计算堆栈上。

Ldc.I4.1 将整数值 1 作为 int32 推送到计算堆栈上。

刚开始的值为 LDC.I4.0 修改为 LDC.I4.1 相当于从 false 变为 true

进去后发现就在底部多了一个 flag

总结: 刚开始不知道 dll 被混淆了,所以一直没有解出来.后来解出来后.因为对软件的不熟悉以及对 IL 指令没有接触过很难进一步解题.浪费了大量的时间.

SUCTF{WeLC0mE_70_5uc7F}