War3小外挂之一键开图

其实，当你对Windwos核心编程有一定的了解后，编写各种游戏小外挂就和写"Hello World！"一样简单。

    War3开图小外挂实现步骤主要有三：1 提升当前外挂进程的访问权限；2 获取War3的进程句柄以及该进程中game.dll的基地址；3 修改War3的进程空间数据(开图)。下面我们对每一部分做详细分析。

**一、 提升当前外挂进程的访问权限**

 Windows系统中，当你想在一个进程中利用OpenProcess获取另一个独立进程的句柄时，你必须提升该进程的权限。要对一个任意进程（包括系统安全进程和服务进程）进行指定了写相关的访问权的OpenProcess操作，只要当前进程具有SeDeDebug权限就可以了。要是一个用户是Administrator或是被给予了相应的权限，就可以具有该权限。可是，就算我们用Administrator帐号对一个系统安全进程执OpenProcess(PROCESS\_ALL\_ACCESS,FALSE, dwProcessID)还是会遇到“访问拒绝”的错误。什么原因呢？原来在默认的情况下进程的一些访问权限是没有被启用（Enabled）的，所以我们要做的首先是启用这些权限。通常我们通过调用OpenProcessToken、LookupPrivilegeValue、AdjustTokenPrivileges等函数来修改进程的访问权限。该部分代码如下：

//提升当前进程的访问权限

**[cpp]** [view plaincopy](http://blog.csdn.net/VictoryKNight/article/details/7580188)

1. **BOOL** ImproveProcPriv()
2. {
3. //得到进程的令牌句柄
4. **HANDLE** token;
5. **if**(!OpenProcessToken(GetCurrentProcess(),TOKEN\_ADJUST\_PRIVILEGES,&token))
6. {
7. MessageBox(NULL,"打开进程令牌失败...","错误",MB\_ICONSTOP);
8. **return** FALSE;
9. }
11. //查询进程的权限
12. TOKEN\_PRIVILEGES tkp;
13. tkp.PrivilegeCount = 1;
14. LookupPrivilegeValue(NULL,SE\_DEBUG\_NAME,&tkp.Privileges[0].Luid);
16. //修改进程权限
17. tkp.Privileges[0].Attributes = SE\_PRIVILEGE\_ENABLED;
18. **if**(!AdjustTokenPrivileges(token,FALSE,&tkp,**sizeof**(tkp),NULL,NULL))
19. {
20. MessageBox(NULL,"调整令牌权限失败...","错误",MB\_ICONSTOP);
21. **return** FALSE;
22. }
23. CloseHandle(token);
24. **return** TRUE;
25. }

**二、 获取War3的进程句柄以及该进程中game.dll的基地址**

提升当前进程的访问权限后，我们就能够轻松地利用OpenProcess获取War3的进程句柄了。首先我们通过进程名称“war3”或者"War3"来获取war3的进程id，然后调用OpenProcess获取war3的进程句柄。

    注意设置外挂进程对war3进程的操作权为：

     PROCESS\_ALL\_ACCESS|PROCESS\_TERMINATE|PROCESS\_VM\_OPERATION|PROCESS\_VM\_READ|PROCESS\_VM\_WRITE

  这样你就能够修改war3进程空间的内容了。该部分代码如下：

**[html]** [view plaincopy](http://blog.csdn.net/VictoryKNight/article/details/7580188)

1. void CWar3OpenMapDlg::InitGameHandle124()
2. {
3. ImproveProcPriv();    //提升当前进程的权限
4. DWORD dwpid = GetProcPIDByName("war3.exe");     //得到war3进程ID
5. if (dwpid == 0)
6. {
7. dwpid = GetProcPIDByName("War3.exe");       //得到war3进程ID
8. if (dwpid == 0)
9. {
10. ::MessageBox(m\_hWnd, "貌似您 没有开war3吧？", 0, MB\_ICONERROR);
11. return ;
12. }
14. }
15. //得到war3进程句柄
16. hopen = OpenProcess( PROCESS\_ALL\_ACCESS|PROCESS\_TERMINATE|PROCESS\_VM\_OPERATION|PROCESS\_VM\_READ|
17. PROCESS\_VM\_WRITE,FALSE, dwpid);
18. if (hopen == NULL)
19. {
20. ::MessageBox(m\_hWnd, "不能打开war3进程！", 0, MB\_ICONERROR);
21. return ;
22. }
23. BOOL bret = FALSE;
24. gameBase = GetDLLBase(\_T("Game.dll"), dwpid);
25. if (gameBase == 0)
26. {
27. gameBase = GetDLLBase(\_T("game.dll"), dwpid);
28. }
29. dSize = 0;
30. gameAddr = (DWORD)GetModuleHandle("game.dll");
32. //设置为内存页为可读可写可执行，(gameBase + 0x01000)为要改变属性的内存起始地址，0x87E000要改变属性的内存区域大小
33. BOOL bVir = VirtualProtect((LPVOID)(gameBase+0x01000),0x87E000,PAGE\_EXECUTE\_READWRITE,&OldProtect );
34. }

**三、 修改War3的进程空间数据（开图）**

前面两部分只是做好了准备工作，到这一步我们才干我们想干的事。需要修改的地址空间可以到网上去查。该部分代码如下：

**[cpp]** [view plaincopy](http://blog.csdn.net/VictoryKNight/article/details/7580188)

1. //hopen：魔兽的进程句柄；i:地址偏移量；w:将被写入到指定进程的地址空间内的数据；l:写入数据的字节数；dSize:可为NULL
2. #define WPM(i,w,l) WriteProcessMemory(hopen,reinterpret\_cast<LPVOID>(gameBase+i),w,l,&dSize)
4. //开图函数
5. **void** CWar3OpenMapDlg::OnButtonOpenmap()
6. {
7. // TODO: Add your control notification handler code here
8. InitGameHandle124();
10. WPM(0x74D1B9,"\xB2\x00\x90\x90\x90\x90",6);
11. ////////////////////////////大地图显示单位
13. WPM(0x39EBBC,"\x75",1);
14. WPM(0x3A2030,"\x90\x90",2);
15. WPM(0x3A20DB,"\x8B\xC0",2);     //用相同指令替换掉
16. ///////////////////////////////显示隐形单位
18. WPM(0x28357C,"\x40\xC3",2);
19. /////////////////////////////////////////////////////显示物品
21. WPM(0x3A201B,"\xEB",1);
22. WPM(0x40A864,"\x90\x90",2);
23. ////////////////////////////////////////////小地图 去除迷雾
25. WPM(0x357065,"\x90\x90",2);
27. ///////////////////////////////////////////小地图显示单位
29. //PATCH(0x361F7C,"\x00",1);
30. WPM(0x361F7C,"\xC1\x90\x90\x90",4);                                             //换了种方法绕过检测
31. /////////////////////////////////////////////敌方信号
33. WPM(0x43F9A6,"\x3B",1);
34. WPM(0x43F9A9,"\x85",1);
35. WPM(0x43F9B9,"\x3B",1);
36. WPM(0x43F9BC,"\x85",1);
38. /////////////////////////////////////////////他人提示
39. //
40. WPM(0x3345E9,"\x39\xC0\x0F\x85",4);
41. ////////////////////////////////////////////////敌方头像
42. WPM(0x371700,"\xE8\x3B\x28\x03\x00\x85\xC0\x0F\x85\x8F\x02\x00\x00\xEB\xC9\x90\x90\x90\x90",19);
43. /////////////////////////////////////盟友头像
44. WPM(0x371700,"\xE8\x3B\x28\x03\x00\x85\xC0\x0F\x84\x8F\x02\x00\x00\xEB\xC9\x90\x90\x90\x90",19);
45. //////////////////////////////////////////////////////资源面板
46. WPM(0x36058A,"\x90",1);
47. WPM(0x36058B,"\x90",1);
48. ///////////////////////////////////////////   允许交易
49. WPM(0x34E8E2,"\xB8\xC8\x00\x00",4);
50. WPM(0x34E8E7,"\x90",1);
51. WPM(0x34E8EA,"\xB8\x64\x00\x00",4);
52. WPM(0x34E8EF,"\x90",1);
53. ////////////////////////////////////////////////显示技能
54. WPM(0x2031EC,"\x90\x90\x90\x90\x90\x90",6);
55. WPM(0x34FDE8,"\x90\x90",2);
57. /////////////////////////////////////////////////技能CD
58. WPM(0x28ECFE,"\xEB",1);
59. WPM(0x34FE26,"\x90\x90\x90\x90",4);
60. //////////////////////////////////////////////资源条
61. //////////////////////////////////////////////野外显血
62. ///////////////////////////////////////////////视野外点击
63. WPM(0x285CBC,"\x90\x90",2);
64. WPM(0x285CD2,"\xEB",1);
65. /////////////////////////////////////////////////无限取消
66. WPM(0x57BA7C,"\xEB",1);
67. WPM(0x5B2D77,"\x03",1);
68. WPM(0x5B2D8B,"\x03",1);
69. //1111
70. /////////////////////////////////////////////////////过-MH
71. WPM(0x3C84C7,"\xEB\x11",2);
72. WPM(0x3C84E7,"\xEB\x11",2);
73. ////////////////////////////////////////////////////反-AH
74. WPM(0x3C6EDC,"\xB8\xFF\x00\x00\x00\xEB",6);
75. WPM(0x3CC3B2,"\xEB",1);
77. WPM(0x362391,"\x3B",1);
78. WPM(0x362394,"\x85",1);
79. WPM(0x39A51B,"\x90\x90\x90\x90\x90\x90",6);
80. WPM(0x39A52E,"\x90\x90\x90\x90\x90\x90\x90\x90\x33\xC0\x40",11);
81. ///////////////////////////////////////////////////分辨幻影
82. }

     外挂在单机和局域网环境下测试通过，只支持1.24e版本。代码可以在本人的资源中下载到。如有任何问题，欢迎交流。