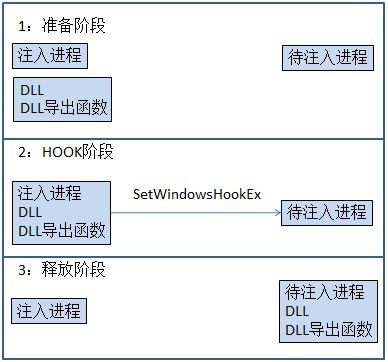
DLL注入技术之消息钩子注入

    消息钩子注入原理是利用Windows 系统中SetWindowsHookEx()这个API，他可以拦截目标进程的消息到指定的DLL中导出的函数，利用这个特性，我们可以将DLL注入到指定进程中。主要流程如下图所示：



**1．准备阶段**  
    需要编写一个DLL，并且显式导出MyMessageProc()函数，主要代码如下：

1. LRESULT WINAPI MyMessageProc(int code,WPARAM wParam,LPARAM lParam)
2. {
3. return CallNextHookEx(NULL, code, wParam, lParam);
4. }

显示导出某个函数需要在.def文件中的EXPORTS填写MyMessageProc()，如下图所示：



**2．HOOK阶段**         
    使用SetWindowsHookEx()之前首先需要将HOOK的DLL 加载到本身的进程中，以此得到DLL的模块句柄，再使用GetProcAddress()得到DLL中显示导出的函数MyMessageProc()的函数地址，最后遍历出待注入进程的线程ID，这样SetWindowsHookEx()就可以利用这些参数进行HOOK了。主要代码如下图所示：

1. //加载DLL到本身进程
2. hMod = LoadLibrary(pDllName);
3. if(!hMod) return FALSE;
4. //得到显示函数的地址
5. lpFunc = (DWORD)GetProcAddress(hMod, "MyMessageProc");
6. if(!lpFunc)
7. {
8. if(hMod) FreeLibrary(hMod);
9. return FALSE;
10. }
11. //得到待注入EXE的进程ID
12. dwProcessId = GetProcessId(pExeName);
13. if(!dwProcessId)
14. {
15. if(hMod) FreeLibrary(hMod);
16. return FALSE;
17. }
18. //得到待注入EXE的线程ID
19. dwThreadId = GetThreadId(dwProcessId);
20. if(!dwThreadId)
21. {
22. if(hMod) FreeLibrary(hMod);
23. return FALSE;
24. }
25. //利用HOOK进行注入
26. hhook = SetWindowsHookEx(
27. WH\_GETMESSAGE,//WH\_KEYBOARD,//WH\_CALLWNDPROC,
28. (HOOKPROC)lpFunc,
29. hMod,
30. dwThreadId);

复制代码

**3．释放阶段**  
    利用LoadLibrary()得到的模块句柄把本身进程的DLL释放掉，代码如下所示：

1. if(hMod) FreeLibrary(hMod);

复制代码

消息钩子注入只需要对SetWindowsHookEx()和DLL导出函数有深刻的认识就可以很容编写，所以代码简单，比较容易实现。