基于远程线程导入API HOOK获取音频数据的原理和实现①

邹山花[1,2]
(1.东南大学软件学院 南京 210096; 2.苏州高博软件技术职业学院 江苏苏州 215163)

摘 要:本文以常用的Windows音频播放API为例,阐述了基于远程线程导入API HOOK获取音频数据的原理和实现方法,解决了一些常用的通讯软件不能实现通话录音的问题。
关键词:线程导入 API HOOK 获取 音频数据
中图分类号:TP319.3 文献标识码:A 文章编号:1674-098X(2009)12(b)-0107-02

在Windows操作系统中,API是指由操作系统提供功能的且由应用程序调用的函数。这些函数分布于不同的DLL文件中或者EXE文件中。应用程序通过调用这些函数来获得一些功能的支持。如音频API函数waveOutOpen、waveOutWrite、waveOutClose等就分布在winmm.dll中。

API HOOK技术是一种用于改变API执行结果的技术。通过API HOOK,可以改变一个系统API的原有功能。基于远程线程导入API HOOK,通常由两个模块实现:一个为Hook安装模块,一般为EXE形式;另一个是Hook执行体,一般为DLL形式。安装模块主要负责向目标进程安装Hook执行体,使得Hook执行体运行在目标进程的地址空间中。Hook执行体则负责搜索到需要的API函数入口点,并改变API函数入口点的地址指向自定义的函数,以便在这些API函数调用之前或之后能做一些我们所希望的工作,如获取音频数据的声道数、采样精度、音频数据等。

一些语音通讯软件如QQ、Skype本身未开放录音功能,如果需要对通话进行录音,则可以采用API Hook的方式获取音频数据。waveOutOpen、waveOutWrite为常用的Windows 音频播放 API,本文以此为例介绍利用远程线程导入API HOOK获取音频数据的方法。

1 Hook安装模块

Hook安装模块向目标进程安装Hook执行体,需要执行如下的步骤:

1.1 获取系统中所有进程的一个快照(hSnapshot)

```
HANDLE hSnapshot=CreateToolhelp32Snapshot(TH32CS_SNAPPROCESS, 0);
```

1.2 然后使用Process32First、Process32Next遍历该快照,获取进程标示号(pid)

```
DWORD pid =0;
PROCESSENTRY32 pe;
pe.dwSize=sizeof(PROCESSENTRY32);
BOOL bOk=Process32First(hSnapshot, &pe);
while(bOk){
if (pe.th32ProcessID!=0 && pe.szExeFile==目标进程文件名){
pid = pe.th32ProcessID;
break;
}
bOk=Process32Next(hSnapshot, &pe);
}
```

1.3 获取目标进程句柄(hProcess)

```
HANDLE hProcess = OpenProcess (PROCESS_ALL_ACCESS, FALSE, pid);
```

1.4 获取LoadLibraryA的函数入口地址(pfnThreadRtn),用以在远程线程中进行调用

```
HMODULE hKernel32=GetModuleHandle("Kernel32");
FARPROC pfnThreadRtn = GetProcAddress(hKernel32, "LoadLibraryA");
```

1.5 在目标进程的虚拟内存区域分配空间,将Hook执行体文件名(strDllPathName,包括绝对路径)写入

```
LPVOID pParaRtn = VirtualAllocEx (hProcess, NULL, MAX_PATH, MEM_COMMIT, PAGE_READWRITE);
size_t iLen = strlen (strDllPathName) + 1;
WriteProcessMemory(hProcess, pParaRtn, strDllPathName, iLen, NULL);
```

1.6 创建远程线程在目标进程中导入Hook执行体

```
HANDLE hThread = CreateReoteThread(hProcess,NULL,0,(LPTHREAD_START_ROUTINE)pfnThreadRtn, pParaRtn, 0, NULL);
```

1.7 待导入完成后,关闭线程、释放内存空间、关闭目标进程句柄

```
WaitForSingleObject(hThread, INFINITE);
CloseHandle(hThread);
VirtualFreeEx(hProcess, pParaRtn, 0, MEM_RELEASE);
CloseHandle(hProcess);
```

2 Hook执行体

Hook执行体在目标进程中搜索到需要的API函数入口点,并改变API函数入口点的地址。

2.1 声明自定义函数

```
MMRESULT WINAPI HookwaveOutOpen(LPHWAVEOUT phwo, UINT_PTR uDeviceID, LPWAVEFORMATEX pwfx, DWORD_PTR dwCallback, DWORD_PTR dwCallbackInstance, DWORD fdwOpen);
MMRESULT WINAPI HookwaveOutWrite(HWAVEOUT hwo, LPWAVEHDR pwh, UINT cbwh);
```

2.2 获取API 函数原来的地址数值

```
HMODULE hmdWinmm=GetModuleHandle("winmm");
PROC g_waveOutOpen=GetProcAddress(hmdWinmm,"waveOutOpen");
PROC g_waveOutWrite=GetProcAddress(hmdWinmm, "waveOutWrite");
```

2.3 获取目标进程中所有模块的快照

```
HANDLE hSnapshot=CreateToolhelp32Snapshot(TH32CS_SNAPMODULE, GetCurrentProcessId());
```

2.4 同样使用Process32First、Process32Next遍历该快照,获取所有模块的输入描述结构列表首地址

```
MODULEENTRY32 me={sizeof(MODULEENTRY32)};
BOOL bOK=Module32First(hSnapshot, &me);
while(bOK)
{
LPVOID pData=
ImageDirectoryEntryToData(me.hModule,TRUE,
MAGE_DIRECTORY_ENTRY_IMPORT, &ulSize);
// 遍历模块的输入描述结构表
bOK=Module32Next(hSnapshot, &me);
}
```

2.5 对获取的输入描述结构列表进行遍历,查找到所需的模块(winmm.dll)的入口地址表(Import Address Table, IAT)

```
PIMAGE_IMPORT_DESCRIPTOR pImportDesc=(PIMAGE_IMPORT_DESCRIPTOR)pData;
for(;pImportDesc->Name; pImportDesc++)
{
PSTR pszModName=(PSTR)me.
```

(下转109页)

①作者简介:邹山花(1972-),女,河南邓州市人,讲师,东南大学软件学院在读研究生,现任苏州高博软件技术职业学院教务处副处长,主要从事教育教学管理、计算机基础教育及计算机网络方向研究。

对于重油炼制废水,BOD与COD的比值远远小于废水可生物降解要求的其最低值,北京大学发现表面流人工湿地对COD,油类物,BOD和TKN表现出很高的去除效率,去除率分别是80%、93%、88%、86%。实验中通过对PH、COD、矿物油、BOD、TKN、TP等指标的测定以及对植物叶子数量、植物高度、植物干重的检测,证明了人工湿地对COD、矿物油、BOD、TKN的去除效果,并总结出了以下结论:1)随着水力负荷的增大,COD的去除率反而减小;2)随着水力负荷的增大,COD去除负荷率随之增大;3)低水力负荷伴随着高植物干重量;4)植物叶子无明显变化,说明油质废水对植物生长影响极小。实验中还发现此人工湿地对TP表现出极差的去除效果,在整个抽样过程中出水TP含量始终高于进水TP,这可能与湿地溶氧及pH值有关,缺乏氧气会导致基质中的金属元素无法吸收p元素,而溶解氧的多少也与湿地植物有关,比如E.crassipes会造成湿地中厌氧状态,E.crassipes和T. domingensis混合使用更有利于吸磷与释磷过程;还有PH值的影响,研究发现PH值过高,基质中Fe元素吸收磷的能力减弱,而Ca吸收能力增强,所以要对废水PH值事先测量然后选取合适的基质。

3 结语

与传统生化工艺相比,人工湿地工艺具有运行简便,抗有机负荷能力强,处理效率高等特点;虽然常规污水处理工艺对油质废水中常见污染物COD、BOD和氨氮具有一定的处理效率,但存在极大的局限性,对金属离子和盐分的去除效果较差,往往影响氨氮的去除效果;湿地不仅在油质废水中得到应用,而且在处理制革废水、采矿废水、农田径流、垃圾渗透液中取得了重大的成果;人工湿地要求水力停留时间较长,处理水量较小,不适宜大中型炼油企业的油质废水的深度处理;人工湿地工艺特别适合我国水处理方面所面临的资金不足、技术含量低的现状,特别对那些缺水或极度缺水的城市地区具有缓解饮用水压力、提高污废水的循环利用率、补充地下水水源等优势。

参考文献

[1] 赖世荣.A/O生物膜法氧化沟工艺在我厂炼油废水处理中的应用[J].江西石油化工,1995,15(3):69～74.
[2] 史毅强.生物膜法氧化沟在炼油废水处理中的应用[J].江西石油化工,1997,11(1):42～46.
[3] 李金成,沈文,王娟,等.两段活性污泥法处理炼油废水的效果及分析[J].青岛建筑工程学院学报,1998,19(2):34～38.
[4] 夏文香,李金成,马书忠.两段活性污泥法处理炼油废水的工艺研究[J].青岛建筑工程学院学报,2000,21(1):52～56.
[5] Zarooni M.A, Elshorbagy W. Characterization and assessment of Al Ruwais refinery wastewater[J]. Journal of Hazardous Material,2006,136(3):398～405.

(上接107页)

```
hModule+pImportDesc->Name;
    if (lstrcmpiA(pszModName, "winmm.dll") == 0)
    {
    LPVOID pIAT=(PBYTE)me.hModule+pImportDesc->FirstThunk;
    // 遍历入口地址表(IAT)
    }
    }
```

2.6 遍历入口地址表,找到匹配的API函数

```
    PIMAGE_THUNK_DATA pThunk = (PIMAGE_THUNK_DATA) pIAT;
    for(;pThunk->u1.Function; pThunk++)
    {
    ppfn=(PROC*)&pThunk->u1.Function;
    if (*ppfn == g_waveOutOpen)
    // 改变waveOutOpen API 入口地址
    if (*ppfn == g_waveOutWrite)
    //改变waveOutWrite API 入口地址
    }
```

2.7 改变API函数入口地址ppfn为自定义函数地址pfnNew

```
    MEMORY_BASIC_INFORMATION mbi;
    VirtualQuery(ppfn, &mbi, sizeof(MEMORY_BASIC_INFORMATION));
    if(VirtualProtect(mbi.BaseAddress, mbi.RegionSize, PAGE_READWRITE, &mbi.Protect))
    {
    PROC pfnNew=(PROC)HookwaveOutOpen;
    WriteProcessMemory(GetCurrentProcess(), ppfn, &pfnNew, sizeof(PROC), NULL);
    }
```

3定义自定义函数

3.1 自定义waveOutOpen函数,获取音频格式

```
    typedef MMRESULT (WINAPI *PWAVEOUTOPEN)(LPHWAVEOUT phwo,UINT_PTR uDeviceID, LPWAVEFORMATEX pwfx, DWORD_PTR dwCallback, DWORD_PTR dwCallbackInstance, DWORD fdwOpen);
    MMRESULT WINAPI HookwaveOutOpen(LPHWAVEOUT phwo, UINT_PTR uDeviceID, LPWAVEFORMATEX pwfx, DWORD_PTR dwCallback, DWORD_PTR dwCallbackInstance, DWORD fdwOpen)
    {
    //自 pwfx 获取音频格式
    return((PWAVEOUTOPEN)g_waveOutOpen)(phwo,uDeviceID,pwfx, dwCallback,dwCallbackInstance, fdwOpen);
    }
```

3.2 自定义waveOutWrite函数,获取音频数据

```
    typedef MMRESULT (WINAPI *PWAVEOUTWRITE)(HWAVEOUT hwo, LPWAVEHDR pwh,UINT cbwh);
    MMRESULT WINAPI HookwaveOutWrite(HWAVEOUT hwo, LPWAVEHDR pwh,UINT cbwh)
    {
    // 自 pwh 获取音频数据
    return ((PWAVEOUTWRITE) g_waveOutWrite)(hwo, pwh, cbwh);
    };
```

4 结语

以上是本人在实际应用中探索出的利用API Hook的方式获取音频数据的实现方法,在此与同行进行交流。API Hook的应用不仅于此,其在屏幕取词、网络防火墙、病毒木马、游戏外挂等方面也有着极其广泛的应用。

参考文献

[1] (美)Johnson M.Hart(安娜等译).Windows系统编程(原书第3版)[M].机械工业出版社,2006(1).
[2] (美)Jeffrey Richter,Christophe Nasarre(葛子昂等译).Windows核心编程(第5版)[M].清华大学出版社,2008(9).
[3] (美)David J.Kruglinski,Scot Wingo, George Shepherd(朱继满等译)Programming Visual C++ 6.0 技术内幕(第五版)[M].北京希望电子出版社,2001(11).
[4] 严佟然.基于VC的HOOK技术实现[J].电脑学习,2004(2).
[5] 周剑岚,冯珊.运用Hook技术实现的软件防火墙[J].华中科技大学学报(自然科学版),2004(3).