文章编号：2096-1472(2018)-07-28-03
DOI:10.19644/j.cnki.issn2096-1472.2018.07.008

基于Windows内核的用户行为收集解决方案

赵晓华，赵树升
(商丘学院应用科技学院理学系，河南 开封 475000)

摘　要：规范企业PC用户的行为使之符合安全性、有用性的要求，借助技术手段收集用户的行为是一种行之有效的方式。分析基于Windows内核技术，实现对用户上网行为、进程操作、文件的读写、删除和重命名进行收集。采用C/S架构，在服务器端文件记录客户端访问的网站信息和进程操作信息，方便对历史记录进行查询，完全基于内核，无进程也无DLL支持，可靠性和安全性高。

关键词：Windows内核；进程操作；上网行为；文件行为
中图分类号：TP309　**文献标识码**：A

A Solution for User Behavior Collection Based on Windows Kernel

ZHAO Xiaohua,ZHAO Shusheng
(*Shangqiu University Applied Science and Technology College,Kaifeng* 475000,*China*)

Abstract: To standardize the behaviors of enterprise PC users to meet the requirements of security and usefulness,it is an effective way to collect users' behaviors by means of technology.Based on Windows kernel technology,this paper implements the collection of user online behaviors and process operation,file reading,writing,deleting and rename. C/S architecture is used to record client's access to web site information and process operation information in server side files,so as to facilitate queries on historical records.It is based on kernel with no process and no DLL support as well as high reliability and security.

Keywords: Windows kernel;process operation;online behavior;file behavior

1 引言(Introduction)

随着计算机在各行各业的重要性日益凸显，如何规范工作人员的行为，提高工作效率已成为各公司机构需要解决的问题。采用强有力的技术手段，实时查看用户上网行为，以及用户进程操作，确保员工高效、安全的使用计算机。

本文基于Windows内核技术，通过采用一系列的技术手段，收集用户上网、进程，以及文件操作行为，可用于分析用户是否有违规行为。

2 系统结构(System structure)

系统结构图如图1所示，由监控客户端、Windows服务器组成，一个服务器可以接收多台PC客户端收集的信息。客户端运行监控程序能够对每个客户端用户的上网行为进行即时的收集。Windows服务器部署在与监控客户端同一局域网，在Windows服务器上运行管理程序。监控客户端程序和Windows服务器的管理服务模块通信采用了C/S架构。下面对这两个功能模块进行简要的介绍。

(1)监控客户端

监控客户端部署在局域网内需要被监控的各计算机上，它的主要功能是收集用户的网络和进程操作行为。

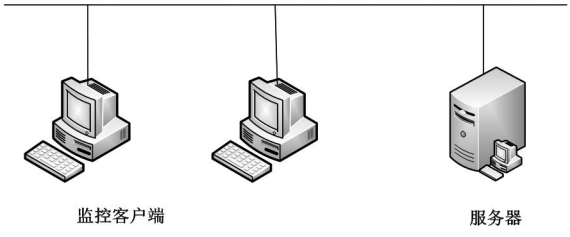

图1 系统结构

Fig.1 System structure

采用进程回调方式捕获进程调用，记录进程创建、进程退出等。采用基于Minifilter(Mini-Filter Installable File System文件系统微过滤驱动)的文件过滤技术，记录文件的打开、读、写、重命名操作。WFP(Windows Filtering Platform Windows过滤平台)对TCP/IP协议栈进行全方位的

过滤拦截，接收发送的TCP数据包，分析用户的网络行为。

为了防止客户端监控程序被非法关闭或异常退出，系统采用一系列自我保护措施。包括注册表回调防止删除注册表启动项、基于IRP((I/O Request Package)的独占方式打开程序、防止加载专业工具删除监控程序。

(2)服务器端

维护所有客户端监控程序的工作状态，采用内核TDI(Transport Driver Interface传输层接口)将客户端收集到的网络和进程操作行为记录到文件，方便对历史记录查询。

3 相关技术概述(Overview of related technologies)

3.1 进程回调

采用进程回调记录进程创建、退出等操作。当一个进程被创建或者删除时，一系列的例程将会被调用。PsSetCreateProcessNotifyRoutine进程回调函数的原型如下：

```
NTSTATUS PsSetCreateProcessNotifyRoutine(
    IN PCREATE_PROCESS_NOTIFY_ROUTINE NotifyRoutine,
    IN BOOLEAN Remove
    );
```

其中，NotifyRoutine表示例程函数的入口地址，Remove为FALSE时，添加例程到链表，为TRUE时，从链表中删除该例程。

PCREATE_PROCESS_NOTIFY_ROUTINE为回调指针，声明如下：

```
VOID(*PCREATE_PROCESS_NOTIFY_ROUTINE)(
    IN HANDLE ParentId,
    IN HANDLE ProcessId,
    IN BOOLEAN Create
    );
```

ParentId和ProcessId用于确定进程，Create参数表明进程是创建还是删除。当Create为True时，例程在新创建的进程的初始化线程被创建后被调用。当Create为False时，例程在进程的最后一个线程被关闭，进程的地址空间将被释放时调用。当进程被删除时，例程运行在进程的最后一个线程中。

3.2 基于Minifilter的文件过滤

MiniFilter相对于传统的sfilter方式，Minifilter编写微小而简单。速度增加的同时，不同软件之间兼容性也得到了提升[1]。

(1)文件过滤中对文件的读写操作进行过滤

注册微过滤器时，填写微过滤器注册结构FLT_REGISTRATION，在FLT_OPERATION_REGISTRATION域中，定义文件打开、文件读IRP_MJ_READ、文件写IRP_MJ_WRITE、文件删除IRP_MJ_SET_INFORMATION、文件重命名等文件操作的回调函数NPPreCreate、NPPostRead、NPPreWrite、NPPreDelete、NPPRERENAME。

(2)获取文件操作信息

在回调函数中NPPreXXX在请求完成之前进行处理，拦截请求本身。在回调数据包FLT_CALLBACK_DATA中包含了请求相关的全部信息，获取到打开的文件名、操作的进程号、文件路径等等。NPPostXXX在请求完成之后，对返回结果进行拦截。

3.3 基于WFP的网络传输信息

WFP由Filter Engine(过滤引擎)、Callout Driver、Filter组成[2]。Filter Engine的内核模式可以对TCP/IP协议栈进行全方位的过滤拦截。Callout Driver是WFP扩展功能的一种机制。Callout Driver由一组Callout函数组成，其中，ClassifyFn函数处理收到的网络数据，例如端口号、ip地址等。Filter包含过滤条件，指定过滤器的action类型、处理action的Callout。

WFP的过滤过程[3]：首先，在过滤引擎(Filter Engine)中注册Callout，在Callout处理函数classifyFn函数中处理数据。然后，初始化Filter，为Filter添加过滤条件，指示哪些数据符合filter的要求，把数据交给Callout处理。处理网络数据包的classifyFn函数，函数原型如下：

```
VOID NTAPI classifyFn(
    IN const FWPS_INCOMING_VALUES0*inFixedValues,
    IN const FWPS_INCOMING_METADATA_VALUES0*inMetaValues,
    IN OUT VOID*layerData,
    IN const FWPS_FILTER0*filter,
    IN UINT64 flowContext,
    OUT FWPS_CLASSIFY_OUT0*classifyOut
    );
```

其中，intFixedValues是指向FWPS_INCOMING_VALUES0_的结构体

```
typedef struct FWPS_INCOMING_VALUES0_ {
  UINT16 layerId;
  UINT32 valueCount;
  FWPS_INCOMING_VALUE0*incomingValue;
} FWPS_INCOMING_VALUES0;
```

(1)获取网络连接的端口号

TCP传输在FWPM_LAYER_ALE_FLOW_ESTABLISHED_V4层建立连接，获取建立连接的信息。

通过inFixedValues->incomingValue[FWPS_FIELD_ALE_FLOW_ESTAB-LISHED_V4_IP_LOCAL_PORT].value.uint16获取发送方端口。

通过inFixedValues->incomingValue[FWPS_FIELD_ALE_FLOW_ESTAB-LISHED_V4_IP_LOCAL_ADDRESS].value.uint32获取

发送方IP。

通过inMetaValues->processId获取进程id。

(2)传输数据信息的获取

在classifyFn处理函数中layerData参数包含了要传输的数据。TCP数据对应的数据类型为FWPS_STREAM_CALLOUT_IO_PACKET。

streamPacket=(FWPS_STREAM_CALLOUT_IO_PACKET*)packet;

streamBuffer=streamPacket->streamData;

获取缓冲区的内容：

RtlCopyMemory(tmpStream,streamBuffer,streamBuffer->dataLength);

此时已经获取了streamBuffer中的内容，对tmpStream进行数据类型转换即可进行数据分析。

3.4 自我保护

在服务器端、客户端安装exe程序执行收集，要防止程序被修改或删除，需要采取一系列的自我保护措施。本文采用三种方式，采用注册表回调方式防止删除注册表启动项，基于IRP实现独占方式打开文件，防止加载专业工具删除程序。

3.4.1 注册表回调防止删除注册表启动项

内核模式驱动程序调用CmRegisterCallback函数注册一个回调，在配置管理每一次注册表操作的信息都会被填充到REG_XXX_KEY_INFORMANTION结构体里。回调例程能够阻止注册表操作[4]。

CmRegisterCallback函数原型如下：

```
NTKERNELAPI NTSTATUS CmRegisterCallback(
  PEX_CALLBACK_FUNCTION Function,
  PVOID                 Context,
  PLARGE_INTEGER        Cookie
);
```

其中，Function是回调函数指针，Context作为回调函数的参数，Cookie是回调的句柄。EX_CALLBACK_FUNCTION的原型如下：

```
EX_CALLBACK_FUNCTION ExCallbackFunction;
NTSTATUS ExCallbackFunction(
  PVOID CallbackContext,
  PVOID Argument1,
  PVOID Argument2)
```

Argument1记录操作类型，Argument2记录有关操作信息的结构指针。若已经注册了注册表回调，当有删除注册表项的操作发生时，Argument1为RegNtPreDeleteKey，Argument2指向一个REG_DELETE_KEY_INFORMATION结构体的指针。此函数如果返回STATUS_SUCCESS，注册表操作就会继续执行，如果返回STATUS_ACCESS_DENIED，注册表操作就不会执行了。

3.4.2 基于IRP的独占方式打开程序

当上层应用程序操作某个设备时，I/O管理器会将I/O请求转化成IRP(I/O Request Package)数据结构对象和一个IRP_STACK_LOCATION数据结构对象数组，IRP中的CurrentStackLocation字符指向IRP_STACK_LOCATION中的某一个元素。IRP_STACK_LOCATION数组中的每一个元素由上一层驱动负责填充。

打开文件函数原型如下：

```
NTSTATUS IrpCreateFile(
OUT PFILE_OBJECT*FileObject,
IN ACCESS_MASK DesiredAccess,
IN PUNICODE_STRING FilePath,
OUT PIO_STATUS_BLOCK IoStatusBlock,
IN ULONG FileAttributes,
IN ULONG ShareAccess,
IN ULONG CreateDisposition,
IN ULONG CreateOptions,
);
```

其中，FileObject是指向文件的指针，FilePath指明文件路径，IoStatusBlock是指向结构体IO_STATUS_BLOCK的指针，当打开文件时，结构体中Information值为FILE_OPENED。DesiredAccess指定访问权限，FileAttributes表明文件对象属性，将文件属性设置为FILE_SHARE_READ共享读。设备和中间层驱动一般设置ShareAccess为0，表示调用者以独占方式打开文件。CreateDisposition指定如果文件存在或不存在时所做的操作。CreateOptions指定驱动创建或者打开文件时需要应用的选项。

3.4.3 防止加载专业工具删除

专业工具强制删除文件，通常基于IRP来完成的：构造IRP，先设置文件的属性，将SECTION_OBJECT_POINTERS结构的DataSectionObject和ImageSectionObject两个域清空，然后删除文件。文件以独占方式打开并且只共享读，即使专业工具使用IRP也不能删除。

3.5 基于TDI的信息上传

基于TDI将用户的进程操作、网络行为、文件读写等信息从客户端内核上传服务器。TDI是一套接口的集合，这套接口连着socket和协议驱动，由协议层驱动实现。传输的步骤如下：

(1)建立并连接TDI

NTSTATUS OpenTDIConnection(char*szIpAddress, unsigned short Port);

szIpAddress表示服务器端ip，Port表示服务器端端口。

(2)客户端传输数据、服务器端接收数据

将客户端机器的mac地址，以及客户端机器的网络行为、文件读写、进程操作等信息发送给服务器端。

unsigned long SendData(char*pData,unsigned long bufferLength);

unsigned long RecvData(char*pData,unsigned long Length)；

其中，pData表示传输的数据指针。

4 结论(Conclusion)

网络行为的结果如图2所示。按行显示网络操作，每行内容为网络操作发生时间、进程号、访问的网络ip、端口号、访问的网址。

进程创建、退出，以及文件的创建、读写操作等收集的结果如图3所示。按行显示操作，每行内容为操作发生时间、标识、操作的进程号、文件路径。其中，标识有进程PROCESS、文件写WRITE、进程创建PCREATE、文件读READ、文件删除DELETE等。

```
20180121l0:12 (0776)131.253.61.66:443->login.live.com
20180121l0:12 (1816)192.168.150.1:2869->192.168.150.1:2869
20180121l0:12 (0776)184.50.87.19:80->ctldl.windowsupdate.com
20180121l0:12 (0776)172.30.65.92:9999->172.30.65.92:9999
20180121l0:12 (1752)111.221.29.253:443->settings-win.data.microsoft.com
20180121l0:12 (1752)111.221.29.254:443->v10.vortex-win.data.microsoft.com
20180121l0:13 (1412)184.50.87.73:80->ctldl.windowsupdate.com
20180121l0:13 (1412)172.30.65.90:9999->172.30.65.90:9999
20180121l0:13 (3996)104.75.250.62:80->cdn.content.prod.cms.msn.com
20180121l0:13 (3996)104.75.240.105:80->tile-service.weather.microsoft.com
20180121l0:13 (3996)104.75.250.62:80->cdn.content.prod.cms.msn.com
20180121l0:13 (3120)202.89.233.100:443->cn.bing.com
20180121l0:13 (0776)104.75.240.105:80->blob.weather.microsoft.com
20180121l0:14 (4604)13.78.94.7:443->mobile.pipe.aria.microsoft.com
20180121l0:14 (4604)104.76.8.237:443->oneclient.sfx.ms
20180121l0:30 (4848)111.221.29.253:443->settings.data.microsoft.com
```

图2 网络行为

Fig.2 Network behaviors

```
20180121l0:11 PROCESS 3388)(SYSTEM)c:\windows\system32\wbem\wmiapsrv.exe
20180121l0:13 WRITE 1464)c:\programdata\microsoft\windows defender\scans\mpenginedb.db-wal
20180121l0:13 PCREATE 3928)c:\windows\system32\userinit.exe-->3996)(Administrator)c:\windows
\explorer.exe
20180121l0:13 PCREATE 3996)c:\windows\explorer.exe-->4596)(Administrator)c:\program files\vmware
\vmware tools\vmtoolsd.exe
20180121l0:13 PCREATE 3996)c:\windows\explorer.exe-->4604)(Administrator)c:\users\administrator
\appdata\local\microsoft\onedrive\onedrive.exe
ator)c:\program files\vmware\vmware tools\vmtoolsd.exe
20180121l0:13 PCREATE 3996)c:\windows\explorer.exe-->4604)(Administrator)c:\users\administrator
\appdata\local\microsoft\onedrive\onedrive.exe
20180121l0:13 READ 4596)\device\harddiskvolume1\program files\vmware\vmware tools\messages\zh_cn
\vmtoolsd.vmsg
20180121l0:13 DELETE 4604)c:\users\administrator\appdata\local\microsoft\onedrive\logs\personal
```

图3 文件、进程操作行为

Fig.3 File & process operation behaviors

本文完全采用基于内核的方式，没有监控程序和服务器端进程。也无DLL支持，可靠性好、安全性高。

参考文献(References)

[1] 谭文,杨潇,邵坚磊.寒江独钓:Windows内核安全编程[M].北京:电子工业出版社,2009.

[2] 黄君胜.基于WFP的终端信息泄漏监控系统的研究与实现[J].计算机应用与软件,2013,30(3):315-318;326.

[3] Windows Filtering Platform[EB/OL]. https://msdn.microsoft.com/en-us/library/windows/desktop/aa366510(v=vs.85).aspx.

[4] ly(cqupt),ljh(cqupt).浅谈基于CallBack的注册表监控和过滤技术[J].黑客防线,2009(7):82-85.

作者简介：

赵晓华(1990-)，女，硕士，助教.研究领域：软件开发.

赵树升(1968-)，男，硕士，副教授.研究领域：安全操作系统内核.

(上接第35页)

录下。然后，启动JBOSS服务。在客户端就可以通过https://10.5.110.199:8443/PKI_JSP/key.jsp访问此服务。浏览器中将显示证书的基本信息。如图7所示。

图7 通过项目显示证书基本信息

Fig.7 Show the basic information of the certificate through the project

6 结论(Conclusion)

EJBCA是一个开源的项目，它具有公开性、开放性和灵活性，它的配置和使用都十分的简单和方便。用户可以通过它自己搭建一个适合自己的CA证书管理中心，从而可以保证用户的信息在网络间传输的安全性。EJBCA是一个十分有用和有意义的开源项目，它对推进PKI技术的发展起到了非常重要的作用。

参考文献(References)

[1] 候梅芳,冯梅.基于PKI的身份认证与访问控制平台的设计[J].微计算机信息,2012,28(1):132-134.

[2] 荆继武,林璟锵,冯登国.PKI技术[D].北京:科学出版社,2008.

[3] 颜海龙,闫巧,冯级强,等.基于PKI/CA互信互认体系的电子政务[J].深圳大学学报(理工版),2012,29(3):113-117.

[4] 陈勤,凌青生,丁宏.安全CA实例_EJBCA的研究[J].计算机工程与设计,2005,26(12):3222-3224.

[5] 周诚,刘电霆.基于EJBCA的CA系统的研究与实现[J].广西轻工业,2009(12):70-71.

[6] 段辉良,周中伟.基于EJBCA的CA系统的应用研究[J].网络安全技术与应用,2008(7):81-82.

[7] 刘博,刘知贵,任立学.PKI认证技术在阅卷系统中的应用与实现[J].计算机安全,2010(05):83-85.

[8] 吴洁明,史建宜.基于EJBCA的Web Services应用研究[J].计算机工程与设计,2013(10):3443-3447.

作者简介：

史建宜(1988-)，女，硕士，工程师.研究领域：信息安全，自动化.

陈新鹏(1990-)，男，硕士，工程师.研究领域：信息安全，数据库.