网络取证完整性技术研究

王文奇,苗凤君,潘 磊,张书钦
(中原工学院计算机学院,郑州市网络安全评估重点实验室,河南郑州 450007)

摘 要: 针对司法取证的要求,结合网络数据的特点,提出了基于网络的动态电子取证模型,描述了总体结构和相关规则.为保证取证网络会话的完整性,设计了基于二维链表的多队列高速网络数据缓存算法,并验证了该算法的有效性,解决了取证模型的关键技术.最后利用插件技术实现了可扩展的取证系统.

关键词: 取证模型; 二维链表; 高速缓存算法; 插件技术
中图分类号: TP393.08 **文献标识码:** A **文章编号:** 0372-2112 (2010) 11-2529-06

The Research on Integrity Technique of Network-Based Forensic

WANG Wen-qi, MIAO Feng-jun, PAN Lei, ZHANG Shu-qin
(*College of Computer Science, Zhongyuan University of Technology, Zhengzhou Key Lab of Network Security Assessment, Zhengzhou, Henan* 450007, *China*)

Abstract: Considering on judicial forensic requirements and the characteristic of network packets, a network dynamic forensic model is proposed, which architecture and related rules are described. An algorithm based two-dimensional linked list and multi-queue which is used to cache network data in high speed network is designed. The effectiveness of the algorithm is analyzed and tested. The algorithm resolves key problem in the above model and ensures the integrality of network session which is saved. Finally, a network forensic system is designed by plug-in, which is extensible and support second development.

Key words: forensic model; two-dimensional linked list; caching algorithm; plug-in

1 引言

要防止入侵,阻止网络犯罪,仅靠诸如防火墙和入侵检测系统等安全产品的过滤和预警功能是远远不够的.由于黑客攻击水平不断提高,网络犯罪呈多样化发展,另外,网络犯罪行为也可能是正常的网络活动,如通过即时通信、Email 等传播非法信息.因此,要从根本上解决网络犯罪问题,就要依靠法律,利用有效的法律手段对黑客行为、对各种各样的网络犯罪予以制裁,这当中关键的问题之一就是取证.网络取证不仅是对于黑客入侵行为的取证,正常的网络行为也可能是需要取证的非法信息.因此,为了能够有效的打击网络犯罪,需要从基于网络的角度来进行电子取证的研究.

近年来国内外在网络取证方面做了许多工作,Case.A 等提出了电子证据的自动发现和关联分析框架 FACE[1],其侧重点是研究证据发现和关联分析技术.Cohen.M 提出了网络取证框架 PyFlag,并分析如何对 HTML、DNS 等类型数据取证[2],其主要研究的是对特定数据的取证.Yongping.T 等提出了一个基于代理的分布式取证模型,其着重研究的是通过代理取证攻击行为以及如何压缩取证的网络数据[3].国内如中科院软件所的孙波等人则从取证机制本身的安全性出发,研究了取证收集系统的保护机制[4].

然而,网络取证目的是提供法庭接受的证据,要求取得的证据是原始的、不可更改的,而网络数据是易逝的,转储过程中可能被修改的.同时网络数据是基于会话的,任何网络数据包的丢失都可能导致会话无法还原.因此,本文将就司法角度如何取证网络数据,并着重对如何保证网络数据完整性方面进行探讨.

2 取证模型描述

本文研究的网络动态电子取证技术适用范围为,在取证人员取得授权的情况下,将取证设施部署于犯罪者最可能经过的网络,利用已掌握的非法信息特征,从网络中过滤出正在实施的非法信息,属于犯罪过程中取证,更有利于及时抓捕犯罪分子,尤其适用于当前网络犯罪.

依据国际计算机证据组织(International Organization on Computer Evidence IOCE)的相关电子取证标准[5,6],结合网络数据的特点,本文提出以下网络取证模型:

定义 1 (网络证据取证模型),定义网络取证模型由以下六元组表示(如图 1 所示):

Net_forensic = {Evidence, Subject, Object, Algorithm_parse, Characteristic, Time}

收稿日期:2009-07-26;修回日期:2010-02-02
基金项目:河南省科技攻关计划(No.082102210082,No.082102210092)

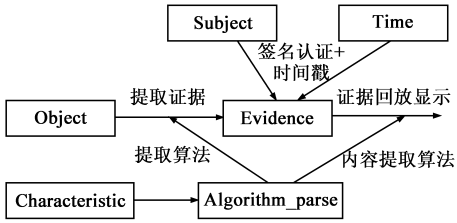

图1 网络取证模型示意图

其中:**Evidence**:表示网络证据的构成,其存在这样一个语义逻辑递进关系:二进制 0、1 代码→网络数据包→网络会话→通信内容.网络证据尽管像普通电子证据一样是由二进制代码组成,但其是由网络数据包构成,而单个网络数据包是没有意义的,需要进而将多个网络数据包重组,在应用层提取包含一定语义的表示非法活动的网络会话.

Subject:取证的主体,具体表征为取证的用户,其对取证的数据负有法律责任.

Object:表示取证的客体,表征为取证的物理设备、设备的物理位置、网络的拓扑结构及设备在其中的逻辑位置.捕获位置一般位于犯罪人员发送网络数据所必须经过的网络节点,如位于网关的交换机和路由器等.

Characteristic:犯罪人员通过网络发送非法信息的特征.网络数据是海量的,不可能保存捕获到的所有网络数据,可以通过提供的非法特征信息,从捕获到的网络数据中提取与犯罪相关的网络证据.如需要监控"法轮功"相关的犯罪证据,犯罪特征为"法轮功".

Algorithm_parse:取证算法集,主要分为两种算法集,其一是网络数据取证算法,根据非法信息特征,判断捕获的网络数据是否有需要取证的信息,如果有需要取证的信息,则将该网络数据包保存,这时需要确保将整个会话保存,否则可能造成信息丢失,还原显示时无法恢复通信内容;其二网络数据还原显示算法,根据网络数据使用的网络协议,将保存的原始二进制网络证据以可视的形式显示出来.不同非法信息采取的网络协议可能不同,分析方法也不同,因此采用的取证算法也不相同,如当犯罪特征为利用 Email 发送非法信息时,则需要根据 SMTP 或者 HTTP 协议将网络数据重组成应用层数据,得到 Email 信息,进而判断是否需要取证.

Time:取证的时间范围,由起止时间和终止时间表示.

设 fe 表示证据集,fs 表示取证主体,fo 表示取证客体,fa 表示取证算法集,fc 表示非法信息特征,ft 表示取证时间段.存在以下规则:

规则 1: $$((fc_1 = fc_2) \rightarrow (fa_1 = fa_2)) \tag{1}$$

即:非法信息特征决定了网络数据取证的算法和网络数据还原算法.本条规则说明需要对不同的网络应用需要设计不同的取证分析和网络数据还原算法,这是由于不同的网络应用对应不同的网络数据格式,需要设计对应的取证分析算法.

$$\forall fs((fo_1 = fo_2) \wedge (fc_1 = fc_2) \wedge (ft_1 = ft_2)) \rightarrow ((fe_1 = fe_2) \wedge (fa_1 = fa_2)) \tag{2}$$

规则 2:即:在相同的取证客体、非法特征以及时间段时,取得的非法证据集是相同的,而与取证的主体无关,也就是与取证人员无关.本规则说明了证据的客观性.

网络证据原始性分析:网络动态证据是由网络数据包构成,它不是具体地存在于某一设备,是动态流动的.取证、复制、保存等过程存在被修改的可能.因此,为保证证据的原始性,由取证主题的私钥对由捕获的原始网络证据和时间构成的时间戳进行签名运算.

普通的电子证据仅要保证原始证据在提取、存储及转移过程中的完整性即可.网络取证则不同,有以下定义:

定义 2 网络取证完整性原则:保存网络数据时,不仅要保证单个网络数据包的完整性,而且要保证作为证据的网络数据包在应用层作为整个会话的完整性.

在网络设备中,网络数据是流动的,而对网络数据的分析判断需要一定的时间,存在一个滞后的效应,一般是网络会话进行中甚至是网络会话结束后,才能根据解析的信息判断网络数据是否需要取证.这时,前面的网络数据包已经丢失,又不可能缓存或保存所有网络数据包,因而容易造成数据包的丢失,破坏网络数据的完整性.因此,有必要在网络取证过程模型中引入攻击预防阶段.需要将网络数据包先缓存起来,而后根据分析的结果,将之前缓存的网络数据包作为证据保存起来,或直接丢弃,通过提前收集并缓存网络数据,以此保证所取得电子证据的完整性.这时,如何缓存网络数据就是一个亟待解决的问题,尤其在高速网络环境下,缓存网络数据的算法更为重要,而且网络取证一般是在网关取证,很可能要面对高速网络的环境.下面将对如何解决这一关键问题进行描述.

3 高速缓存算法设计

3.1 需要解决的问题

高速环境下,缓存网络数据需要解决以下问题:**(1)存储空间问题**.如何能够获得超大存储空间,存储尽量多的网络数据,存储的数据越多,网络数据缓存的时间就越长,取证原始证据时就越可能完整;**(2)缓存算法的空间效率**.缓存网络数据就需要给每一个网络数据包分配缓存空间,而网络数据的大小不定,如果为每一个数据包分配相同的空间,则只能按照最大数据

包占用的空间分配,缓存数据量较大时势必浪费大量的缓存空间.但是如果根据数据包大小分配相应的空间,则无论是首次适应算法、最佳适应算法、还是最坏适应算法,频繁申请和释放网络数据所需的空间时,会造成分配效率极低;(3)**缓存算法的时间效率**.在海量网络数据中,需要保证在缓存空间中数据包遍历时间复杂度为常数级,因为在高速网络环境下需要频繁地进行插入、删除等操作,即使算法时间复杂度为线性,也可能造成网络数据包的丢失.

3.2　超大内存的获取技术

为尽量减少磁盘 I/O,缓存数据采用物理内存缓存.对于 32 位应用程序 Windows 环境下,应用程序只能访问 2GB 的进程地址空间,可以采用微软提供的 AWE (Address Windowing Extensions)技术获取超大内存[7].使用 AWE 技术,可以支持 64GB 的物理内存.对于 64 位的 Windows 操作系统可以支持 128GB 物理内存,则无需特别处理.

3.3　高速缓存算法设计

根据网络数据的特点,结合网络电子取证需要,设计了基于二维链表的多队列高速数据缓存算法,目的是解决网络数据高效缓存和网络会话高效遍历问题.

3.3.1　缓存网络数据包的存储结构

有研究资料表明[8,9],首先,互联网中 40%以上的网络数据包都是 100 字节以下的网络数据数据包,这是由于网络中含有大量对网络进行各种控制的网络数据包,如建立、关闭连接,了解当前网络状况等;其次,网络中也包含了大量的网络所能传播最长字节数的网络数据包,这是由于在网络传送大数据时,总是按照其最大传输能力传输网络数据,尤其网络中大量的网络活动是浏览网页、传输文件以及利用 P2P 技术下载或观看电影等,这些网络活动传送的网络信息都不是一次传输所能完成[10].目前,以太网是最广泛使用的局域网,网络上最大网络数据包一般为 1500 字节链路层数据包.

针对这一特点,我们设计了 3 种固定大小存储单元类型(表 1 所示),分别为 100 字节、1000 字节和 1500 字节.网络数据包根据其大小分别缓存于不同类型的存储单元中.

表 1　存储单元与保存网络数据包

存储单元(字节)	保存网络数据包(字节)
100	< 100
1000	100 ~ 999
1500	1000 ~ 1500

为保证取证的网络数据是完整的网络数据,应按照会话保存,采取的原则是要么把该会话的所有网络数据包全部缓存下来,要么不保存该会话的网络数据.为此设计了 3 个基于二维链表的多队列存储结构,分别用于缓存表 1 中不同大小的网络数据包,每个存储结构如图 2 所示.

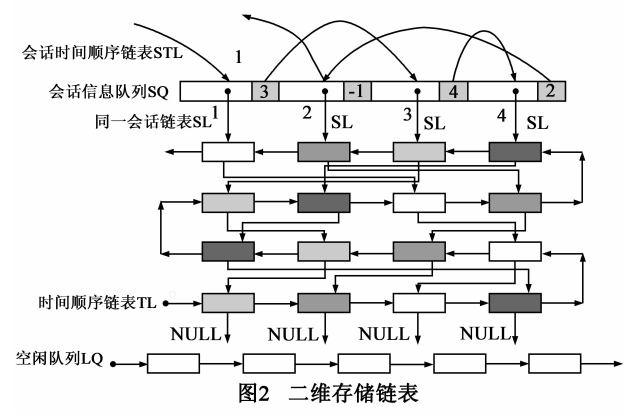

图2　二维存储链表

基于队列的二维链表存储结构由以下链表和队列组成:

(1)**空闲队列 LQ**　将没有分配网络数据的存储单元连接起来,当需要缓存网络数据时,从空闲队列取出存储单元,缓存网络数据.

(2)**会话信息队列 SQ**　该队列中保存了每个会话的信息,由源 IP 地址、源端口、目的 IP 地址、目的端口、协议等唯一标识,并指向二维链表中一个同一会话链表 SL.会话信息队列的存储空间是一个连续的存储空间,目的是按照哈希函数直接得到会话在 SQ 中的位置,哈希函数随后介绍.为便于得到 SQ 中会话的时间顺序,由会话时间顺序链表 STL 将会话信息按照时间的先后顺序链接起来.

(3)**二维链表**　每个包含网络数据的存储单元位于二维链表中,一维链接所有同一会话的网络数据(图 2 中 SL),头指针为会话信息队列 SQ 中的指针,尾指针为 NULL,目的是利用链表得到同一会话的网络数据.而另一维链表则是一个时间顺序链表(图 2 中 TL),把所有的缓存网络数据按照时间顺序链接起来,目的是当空闲队列 LQ 中没有可使用的空闲单元时,可以按照先进先出的原则把保存最久的网络数据缓存单元释放出来.

3.3.2　缓存算法描述

为避免频繁地申请释放缓存空间,需要首先申请一个大缓存空间,然后将申请的存储空间初始化,按照存储单元大小不同将存储单元链接到对应的空闲队列 LQ 中.

网络数据包缓存过程描述如下:

(1)当有网络数据需要缓存时,根据数据包的大小在对应的空闲队列 LQ 中分配存储单元,当空闲队列还有存储单元时,直接获得存储单元;否则,从时间链表 TL 中得到保存最久的数据包,并将该数据包所在会话

的所有存储单元回收到空闲队列 LQ,然后从空闲队列 LQ 中申请存储单元.

(2)利用哈希函数在会话信息队列 SQ 中查找是否已经保存该会话的网络数据,如果有,则直接插入该会话链表,转至(4);否则,产生一个新的会话链表 *SL* 插入队列,如果 SQ 队列已满,转至(3),否则转至(4).

(3)按照先进先出的原则删除保存最久的会话网络数据,并把释放的缓存单元回收到空闲队列 LQ.

(4)将存储单元在时间链表 TL 尾部插入.

3.3.3 哈希函数设计

算法的操作主要集中于网络数据包的插入、删除以及网络数据包的保存等操作.而操作过程是基于会话进行的,由于存储结构为链表结构,对链表操作的时间复杂度一个常数,因此,以上操作的时间复杂度主要集中于在会话链表 SQ 中如何尽快定位,它直接决定了缓存算法所需的时间.

在高速网络环境下,超大缓存空间中,缓存算法的时间复杂度要求达到常数级,否则可能造成数据包的丢失.为此需要设计哈希函数实现定位操作.哈希函数的选择决定了会话信息队列 SQ 中已保存的会话分布是否均匀,映射到 SQ 时,冲突的概率是否足够小,查询操作能否足够快地定位.

哈希函数的设计基本上只限于 5 个域:目的/源 IP 地址、目的/源端口号、协议.下面首先分析它们的分布规律.

文献表明[11],IP 地址的高位部分即网络地址部分比低位部分即主机地址部分更为稳定,IP 地址中低 16 位更富于变化.取证系统针对某一网段监控取证时,取证的网络一般不可能分配了低 16 位的所有地址空间,而一般集中于几个子网段内,在 16 位中最低的 8 位变化性更大.但是在实际的取证过程中,综合考虑源、目的端 IP 地址.

网络层协议域的取值是 0 ~ 255 中很少几个值,目前协议域绝大部分的取值为 TCP、UDP、ICMP、IGMP、IPINIP 等.目的端口和源端口的取值往往是 0 ~ 65535 中极少的一部分.所有端口被分为两类:一类是保留端口,端口号为 0 ~ 1023;另一类是临时端口,端口号大于等于 1024,一般为客户端端口,通常客户端端口号的生成与操作系统相关,其大于 1024 而规律性较差,但是客户端在和远程主机多个会话时,其客户端端口号一般是连续的,也即表示端口号的二进制字段中低位更富于变化.目前,服务器端的保留端口通常是 80(TCP)、20/21(FTP)、25(SMTP)以及 QQ 等常用应用程序的端口号.因此,实际的网络数据包中,端口和协议域取值的不同情况组合是有限的.

综合以上网络数据包特点分析,哈希函数设计如图 3 所示.

首先对网络数据包的端口号进行哈希运算,其中小于 1024 的端口号按图 3 所示算法映射到 0 ~ 7 的数字,并转化为三位二进制前面补 0 成 4 位,对于大于等于 1024 的端口号转化到二进制,取其后三位后前面补 1 成 4 位.源、目的端口号采用相同运算,并将运算的结果进行异或运算,最后结果作为哈希函数结果的 0 ~ 3 位;网络数据包中源 IP 地址和目的 IP 地址的最低一个字节进行异或运算后作为哈希函数结果的 4 ~ 11 位.网络数据包的协议按图 3 所示算法映射到 0 ~ 4 的数字,并转化为二位二进制后作为哈希函数结果的 12、13 位.将网络数据包中源 IP 地址和目的 IP 地址低第三个字节进行异或运算后作为哈希函数结果的 14 ~ 21 位.最后根据会话信息队列 SQ 的大小,取哈希函数结果后 n 位的十进制值作为网络数据包在会话信息队列 SQ 中的位置,因此,一般设置会话信息队列 SQ 的大小为 2^n.把源、目的 IP 地址最低两个字节在哈希函数结果中设置于不同的位置原因在于:IP 地址的最后一个字节差异性更大,同时由于会话信息队列 SQ 大小的限制,会舍弃哈希函数结果的前几位.哈希算法中之所以选择异或运算除其较为简单外,还可以在 4 种不同的操作数组合中,产生 2 个"1"和 2 个"0",即其运算结果也是均匀分布的,这是其他操作,如与、或等所不能达到的.

哈希表采用的解决冲突方法为:当通过哈希表产生的结果发生冲突时,向后顺序查找.最大查找长度 L 设为 $2^n/(10^3 \sim 10^4)$,这里 2^n 为会话信息队列 SQ 的大小.当查找长度达到 L 时,如果是保存新的网络数据,则认为会话信息队列已满,按照最久先出(the Oldest First Out OFO)的算法删除最久的网络会话信息,利用会话时间顺序链表 STL 中得到缓存最久的网络会话,并将该会话删

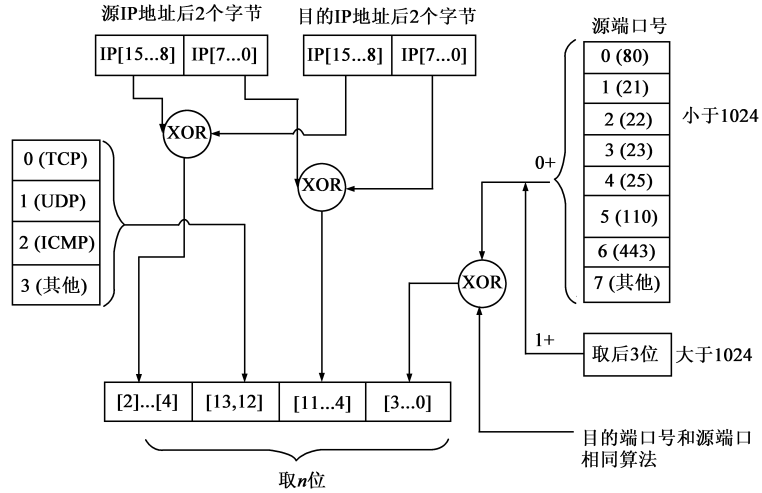

图3 哈希函数算法示意图

除,按照上述哈希函数重新计算查找结果,如果没有找到,则重复计算直至得到存储位置;如果是在 SQ 中查找会话信息,则认为返回 SQ 中没有该会话.

3.3.4 保存数据算法

当需要保存缓存的网络数据时,是按照会话保存,需要保存同一会话的所有网络数据.这样保存的数据包时,尽管可能破坏了不同会话数据包之间的时间顺序,但是由于对网络数据的分析、还原等都是基于会话的,所以按照会话保存原始数据并没有破坏网络数据的完整性.

由上述插入网络数据包算法可知,由于缓存时在每个会话链表 SL 中都是顺序插入的,每个会话链表是按照时间顺序链接的.但是同一会话的网络数据包大小不一样的,可能缓存于不同存储单元的会话链表 SL 中,如果简单地直接顺序存储位于不同存储单元的会话链表 SL 中网络数据,可能破坏了原来网络数据包的时间顺序.需要采用一定的算法恢复同一会话网络数据包的时间顺序,采用算法如图 4 所示.

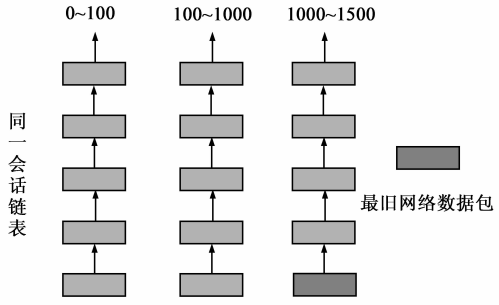

图4　缓存同一会话链表

根据保存的会话信息从 3 个存储单元会话信息队列中找出相应的会话链表,由于每个链表都是时间上顺序保存,每个链表的头部就是该链表所有网络数据包中最早的数据包,因此可以从 3 个链表的头中,比较得到缓存最久的网络数据包,保存到对应的证据库中,在链表中删除该数据包,并继续在 3 个链表查找缓存最久的网络数据包,如此重复,直至 3 个链表都为空.从而可以按照原来网络数据包的顺序将整个网络会话保存.

3.4 算法分析

(1)存储能力　算法中网络数据的缓存能力与两个参数有关,其一是与申请的缓存网络数据内存空间大小,申请的内存空间越大,缓存的网络数据包就越多,其二是与会话信息队列 SQ 有关,缓存网络数据是按照会话缓存,当会话信息队列 SQ 为满时,需要删除旧的会话,即缓存能力存在以下公式:

$$\text{缓存能力} = \{\text{数据包个数 PN, 会话数 SN}\} \subset \{PN < PNmax\} \wedge \{SN < SNmax\} \quad (3)$$

经过大量的测试,会话信息队列 SQ 的大小 SNmax 与网络数据包个数 PNmax 应保持以下关系:

$$\lambda = SNMax : PNmax = 1 : (10 \sim 30) \quad (4)$$

其中,申请的缓存空间越大,λ 值越小.

(2)时间复杂度分析　由于采用哈希函数直接定位会话位置,网络数据在缓存中的插入、删除和保存等操作都是对链表的插入和删除操作,因此对网络数据包操作的时间复杂度为常数级.

3.5 算法测试

将取证系统运行于中原工学院校园网的出口处,交换机为千兆以太网交换机,流量最大为 750Mb/秒左右.取证主机的内存采用 DDR2 内存,配置 16G,CPU 采用双核 CPU,主频 2.5G.

图 5 表明了流量与平均会话缓存时间关系图.由图可知在申请 8G 的内存时,可以在 400MB/s 的网络环境下,可以使网络数据包平均缓存时间达到 15s 以上,基本上满足一般取证分析系统中所需的分析时间.

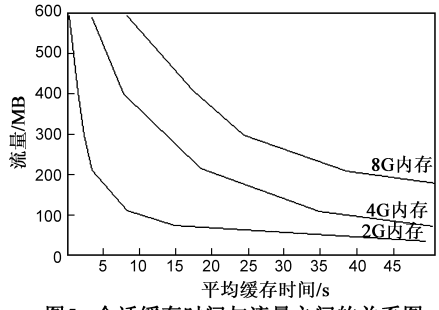

图5　会话缓存时间与流量之间的关系图

从上述测试也可以看出,在一般的高速流量下单机缓存网络数据是可行的,而在超高速流量(1Gbps 以上)下,单机缓存就无能为力.参考许榕生等提出的基于集群方式分析处理高速网络数据包[12],为解决超高速数据缓存提供了一个解决方案,这也是我们未来工作的研究方向之一.

4 基于插件的取证分析技术

非法特征信息是多种形式的,从而决定了网络取证模型的算法集包含多种算法,这就要求网络取证分析算法是动态的、可扩展的.但是电子取证过程要求符合司法取证的要求,需要保证取证数据的原始性,要求证据处理过程是可控、可认证的,这就要求网络取证是稳定的.为解决这一矛盾,采用了取证分析算法与网络数据处理相分离的原则.在程序设计上,可以采用网络数据处理作为主程序框架,功能包括网络数据包的捕获,以及根据分析的结果对包括整个会话在内的网络数据包加密签名、保存等,以保证证据的原始性完整性.网络证据的分析可以采用插件技术,以达到动态可扩展目的.插件采用 DLL 技术,不同类型的网络数据分析过程在 DLL 中实现,根据取证命令对网络数据包进行分析,判断是否对网络数据取证,也可以对不同形式

的网络数据还原显示.在公开编程接口的情况下可以使取证系统具有二次开发能力.

目前程序设计基于 Visual Studio 2005 开发,主要实现了特定主机取证、邮件取证以及黑客入侵取证,其中黑客入侵取证过程是,当入侵检测系统检测到黑客入侵行为后,发出取证命令,取证插件通知主程序对相应的网络数据包取证,达到入侵分析和取证分离的目的,从而也可以更好地利用已有成熟的入侵检测产品.入侵检测产品一般存在一定的漏警率,为此,当入侵检测系统检测某一主机有攻击行为时,则利用缓存算法保存该主机的所有网络数据,分析时再进一步分析,这样在一定程度上降低了漏报警的可能,而对于没有报警事件的主机,则无法取证,实际上如何分析入侵行为是入侵检测分析技术研究的内容.

5 结论

网络取证是近年取证技术研究的热点,但是目前取证研究集中于对黑客攻击的分析及分析结果的取证,或者是针对某一特定网络活动的取证.本文在分析已有研究成果的基础上,结合网络取证的特点和司法证据的要求,提出了网络动态取证的模型,并给出了规则描述和取证原则,设计了基于二维链表多队列缓存网络数据的算法,解决了如何保证网络会话完整性这一关键性问题;基于该取证模型,利用基于插件技术实现了取证系统,并具有良好的可扩展性,能够适应不同类型的网络取证需求.从而验证了模型的可行性.

参考文献:

[1] Case A, Cristina A, Marziale L, Richard G, et al. FACE: Automated digital evidence discovery and correlation[J]. Digital Investigation, 2008, 5: S65 – 75.

[2] Cohen M, Pyflag- An advanced network forensic framework [J]. Digital Invest- igation, 2008, 5: S112 – S120.

[3] Yongping T, Thomas E. Daniels. A simple framework for distributed forensics[A]. Second International Workshop on Security in Distributed Computing Systems (SDCS) [C]. Columbus: IEEE Press, 2005. 163 – 169.

[4] 孙波,孙玉芳,张相锋,梁彬.电子数据证据收集系统保护机制的研究与实现[J].电子学报, 32(8):1374 – 1380.
Sun Bo, Sun Yufang, Zhang Xiangfeng, Liang Bin. Research and implementation of the protection mechanism for digital evidence collecting system[J]. Acta Electronica Sinica, 2004, 32(8):1374 – 1380. (in Chinese)

[5] IOCE. http://www.fbi.gov/hq/lab/fsc/ba ckissu/april2000/swgde.htm[OL]1999-11-8/2009-2-3.

[6] IOCE. Draft Best Practices on the Examination of Digital Evide [OL]. http:// www.ioce.org /fileadmin/user_upload/2002/Guidelines% 20for% 20Best% 20Practices% 20in% 20Examination% 20of% 20Digital% 20Evid. pdf, 2002-5-6/2009-2-3

[7] Microsoft. Address Windowing Extensions[OL]. http://msdn.microsoft.com /en-us/library/aa366527(VS.85).aspx. 2009-1-15/2009-2-3

[8] Apisdor J, Claffy K, Thompson K, et al. OC3MON: Flexible, affordable, high performance statistics collection[A]. Proc of Internet Society's 7th Annual Conference[C]. Kuala Lumpur: Internet Society, 1997. 97 – 112.

[9] Claffy K, Miller G, Thompson K, The nature of the beast: Recent traffic measurements from an Internet backbone[A]. The Eighth Annual Conference of the Internet Society (INET'98)[C]. Geneva, Switzerland, 1998. 21 – 24.

[10] 李玉峰,邱菡,兰巨龙,杨建文.核心路由器转发引擎缓存需求分析[J]. 电子学报,2008,36(7):1421 – 1428.
Li Yufeng, Qiu Han, Lan Julong, Yang Jianwen. An analysis of memory demand for forwarding engines in core routers[J]. Acta Electronica Sinica, 2008, 36(7): 1421 – 1428. (in Chinese)

[11] Talbot B, Sherwood T, Lin B. IP Caching for Terabit Speed Routers[R]. Rio de Janeiro: IEEE Computer Society Press, 1999. 1565 – 1571.

[12] 杨彬,李雪莹,陈宇,许榕生.利用 LINUX 集群实现高速网入侵检测[J].计算机工程与应用,2003,39(23):151 – 153.
Yang Bin, Li Xueying, Chen Yu, Xu Rongsheng. implement high-speed network Intrusion detection via Linux cluster[J]. Journal of Computer Engineering and Applications, 2003, 39(23):151 – 153. (in Chinese)

作者简介:

王文奇 男,1971 年生于河南安阳,副教授,西北工业大学工学博士,研究方向为电子取证、入侵检测等.
E-mail: ww7109@163.com

苗凤君 女,1970 年生于辽宁昌图,副教授,华中科技大学博士生,研究方向为电子取证、网络存活技术等.

潘 磊 男,1975 年生于河南郑州.讲师,哈尔滨工业大学硕士,研究方向电子取证,安全评估技术等.

张书钦 男,1978 年生于河南禹州,副教授,哈尔滨工业大学博士,主要研究方向为网络安全、可信计算.