基于 Windows 主机日志的取证分析方法研究

申月莉

(太原工业学院计算机工程系，山西太原 030008)

摘 要: 本文用于实现一种基于 Windows 主机日志的取证分析方法. 提出了基于 EventID 分类模型的冗余数据清理技术、基于 FP_Growth 的日志分析算法 PFP_Growth、基于模拟攻击的格式化规则匹配方法以及基于规则库与属性跟踪的场景重构方法. 经过实验证明了 PFP_Growth 算法在日志分析方面的高效性和重构方法的有效性.

关键词: 计算机取证; PFP_Growth 算法; 格式化规则匹配; 场景重构

中图分类号: TP391 文献标识码: A 文章编号: 1009 - 4970(2016) 08 - 0062 - 06

DOI:10.16594/j.cnki.41-1302/g4.2016.08.016

0 引 言

日志文件记录了操作系统和应用程序以及用户的操作过程，同时也记录了入侵行为留下的痕迹. 日志文件在分析入侵行为、提取入侵证据方面有着举足轻重的地位. 为了提取被攻击主机当中的入侵证据，重构出入侵过程，还原入侵场景，本文对日志文件进行研究，根据日志记录挖掘关联规则，并进一步重构入侵场景.

挖掘关联规则的算法有很多，比如 Apriori 算法、FP_Growth 算法等[1-3]，这些算法在对大量的日志记录进行挖掘时产生的关联规则意义不明确，且效率较低. 本文结合日志记录的特性，提出了一种新的挖掘算法.

1 日志文件获取与格式化存储

Windows 操作系统包含的日志有系统日志、防火墙日志、应用程序日志、安全日志、WWW 日志、FTP 日志、DNS 日志、Schedule 服务日志、文件目录日志等. 日志文件对于监控系统中事件记录的主要作用有: 审计用户行为、报告可疑行为、作为电子证据、确定入侵范围、恢复系统等.

Windows 操作系统的日志由事件记录组成，事件记录包括记录头、事件描述、附加数据. 日志信息可以通过事件查看器打开，也可以通过打开日志文件查看. 事件记录结构见表 1.

表 1 事件记录结构

记录头	日期	时间	主体标识	计算机名
	事件标号	事件来源	事件等级	事件类别
事件描述	事件描述区的内容取决于具体的事件，可以是事件的名称、详细说明、产生该事件的原因、建议的解决方案等.			
附加数据	可选数据，通常以 16 进制显示.			

Windows 系统中的日志主要分为两大类: 一类是以二进制方式存储的受保护的文件; 另一类是以文本方式存储的文件. 对于二进制方式存储的日志文件，本文主要针对受 Eventlog 保护的日志文件; 对文本方式存储的文件，本文主要研究防火墙日志.

EventLog 类是微软针对. NET 平台提供的获取日志文件的函数，在写入事件记录之前要先创建事件日志源. EventLog 类提供了 12 个公共属性、3 个受保护的属性、24 个公共方法、4 个受保护的方法以及两个公共事件.

FileSystemWatcher 类是微软为. NET 平台提供的监听文件改变的函数. 通过该函数可以监控到日志文本文件的变化. 对日志文件的处理和对 Eventlog 处理方法类似.

收稿日期: 2016 - 03 - 05

作者简介: 申月莉(1989—) ，女，山西霍州人，硕士，助教. 研究领域: 信息安全

日志文件的每一条记录包含了事件发生的详细信息. 每种类型的日志文件所采用的格式都不相同. 在分析日志时，有些属性非常重要，有些属性不是很有必要. 另外，日志记录的数据量庞大，如果分析所有日志记录，工作量会很庞大，会对分析造成很大困难，从而降低分析过程的效率. 所以，根据分析需求，对日志进行规范化管理，将日志统一表示，选取有利的日志字段，形成格式化日志. 通过对日志记录的研究，EventLog 日志格式化标准和文本日志格式化标准见表 2.

表 2　日志格式化标准

日志类型	属性
Eventlog 日志	Date，time，EventID，Eventtype，category，user，description
防火墙日志	Date，time，action，size，src-ip，src-port，dst-ip，des-port，protocol

其中，EventLog 日志中的 Description 属性，对于某一特定的 EventID，它所包含的字段有很多相同的信息，对于不同的 EventID，它所包含的字段各不相同，而这些信息对于入侵分析非常重要. 因此，需要对该字段进行信息再提取，表 3 为针对特定 EventID 的 Description 属性字段再提取的信息描述.

表 3　描述字段信息再提取部分描述

EventID	Description 提取信息或者属性组
680	尝试登录的用户；登录账号；源工作站、错误代码、登录类型
517	安全 ID、账户名称、域名
4634	帐户名、账户域、登录 ID、登录类型
4625	新过程 ID、映像文件名、创建者过程 ID、用户名、登录 ID
903	软件保护服务已经停止.
1530	Windows 检测到注册表文件仍在由其他应用程序或服务使用. 将立即卸载此文件.
7036	xxxxx 服务处于 停止 状态
12	操作系统已在系统时间 20xx－xx－xxTxx：xx：xx. xxxxxxxxZ 启动

2　PFP_Growth 日志分析算法

在获取日志文件并对日志文件进行格式化存储之后，下一步的工作就是对日志文件进行挖掘分析. 根据日志文件的特点以及攻击行为对日志文件的影响特征，在对日志文件进行分析之前首先需要进行冗余数据清理. 本文提出了基于 EventID 分类模型的冗余数据清理技术.

该技术的核心思想是在对日志进行统一分析之前，先对日志中的记录进行分类汇总，剔除不需要分析的记录，从而达到清理冗余数据的目的. EventID 分类模型依赖于 EventID 所代表的含义，通过分析 EventID 所代表的含义，判断该条记录的意义，从而发现系统的改变以及影响系统安全的因素. 在 EventID 分类模型中，将其分为三个数据库，分别是：关注 EventID 库、不关注 EventID 库和未知 EventID 库. EventID 分类模型如图 1 所示. 在该模型中，将确定与已知攻击相关的 EventID 放入关注 EventID 库中；将确定与已知攻击不相关的 EventID 放入不关注 EventID 库；将不明确的 EventID 放入未知 EventID 库. 在分析的过程中，通过将各类 EventID 库逐步与入侵行为库匹配，以不断更新各 EventID 所归属的库. 在后续的分析过程中，针对关注的 EventID 所在的日志记录进行深入分析，从而能减少后续挖掘的工作量.

本文在数据库初始化步骤中，根据前人的经验并结合作者实验分析和观察进行归纳总结，得出关注 EventID57 个，不关注 EventID82 个，未知 EventID15 个.

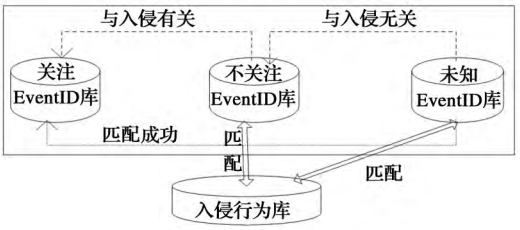

图 1　EventID 分类模型

在对日志文件进行挖掘分析之前，经过对日志记录中包含的属性字段以及属性字段的值进行分析，发现采用挖掘算法进行分析存在以下问题：

(1) 日志记录的每个属性有大量的属性值[4]，而且不同的属性可能有相同的值；

(2) 特定属性的属性值记录得过于精细，如时间属性. 在分析的过程中并不需要精确到秒；

(3) 存在一些属性参与全程挖掘分析步骤，但是在某些分析步骤中并不关注该属性的相关规则；

采用 FP_Growth 算法对日志记录进行挖掘时，生成 FP_Tree 的过程中数据库中每个事务包含的项目均要参与运算，因此生成的 FP_Tree 太过庞大，

使得后期的频繁模式挖掘比较困难[5-6]. 因此，本文针对日志分析时面临的问题，结合日志记录和入侵行为的特征，提出了一个基于 FP_Growth 的日志分析改进算法——剪裁频繁模式增长算法(Prune Frequent Pattern_Growth 算法)，简称 PFP_Growth. 算法的主要步骤如下:

第一步: 将日志记录中的时间属性划分为 24 个时间段[7-9]，如此可以将时间属性的属性值范围从 00:00:00 到 23:59:59 缩减为 0 到 23.

第二步: 在属性值前添加“属性名: ”，从而对所有的属性进行前缀属性意义处理，避免了同一属性值可能归属于不同属性的困扰. 如 23 可能是时间属性值，也可能是端口属性值; 192.168.56.27 可以是源 IP 地址，也可以是目的 IP 地址. 经过处理以后的日志挖掘产生的频繁项集可从{ <192.168.56.27，192.168.56.56: 2 >} 变为{ <SIP:192.168.56.27，DIP:192.168.56.56: 2 >}，含义较为明确，有利于挖掘分析.

第三步: 设置约束属性组(Constraint Properties Group)，即将需要进行挖掘的属性或者特定的属性值设置为约束属性组. 如{ “Time: [20 - 23]”，“Action: [Drop、Open]”}.

第四步: 扫描事务数据库 TD[10]，首先，通过对每个事务中包含的项按照约束属性组进行筛选，保留约束属性组中涉及的属性，以实现数据的剪裁，并构成剪裁事务数据库 PTD; 其次，重新扫描 PTD，以得到项的集合以及在数据库中出现的频度，并将满足最小支持度阈值 Min_Support 的项保留，以得到频繁 1 - 项集; 最后，对频繁 1 - 项集中的项按照频度降序排列，并生成项目头表，记为 Items_Head_Table.

第五步: 按照 Items_Head_Table 中项的先后顺序对 PTD 中事务包含的项进行重新排列，创建 PFP_Tree 的根节点 T，记 PTD 中事务包含项列表为 [p|P]，其中 p 是第一个项，P 是列表中除 p 以外的剩余项. 通过调用 Insert_Tree([p|P],T) 将事务添加到 PFP_Tree 中. Insert_Tree([p|P],T) 的实现方法是: 如果 T 为 Null，创建一个新节点 N 作为 T 的子结点，初始化 N. Count 为 1，并且将 N. Link 和那些具有相同 Item_name 的结点链接起来; 否则，如果 T 有一个子结点 C，且 C. Item_name = p. Item_name，则将 C 的 Count 加 1，之后递归调用 Insert_Tree(P,T) 直到 P 为空. PFP_Growth 算法伪码如表 4 所示.

表 4 PFP_Growth 算法伪代码

```
输入: 日志记录数据库 Log_DB，约束属性组 Con_Attributes Ⅰ，
      项目表头 Items_Head_Table Ⅰ
输出: 频繁项集 Frequent_Item
1. procedure PFP_growth( Tree，a) {
2. for Log_DB 中的所有记录
3. for 约束属性组 Con_Attributes 中的所有属性
4. 统计所有满足约束条件的属性值出现的次数;
5. for 约束属性组 Con_Attributes 中的所有属性
6. if ( Ai. count > = Min_Support)
7. Items_Head_Table [i] = Ai;
8. 将 Items_Head_Table 中的所有属性值按照计数递减排序;
9. if PFP_Tree 包含单个路径 P then{
10. for 路径 P 中结点的每个组合( 记作 b)
11. 产生模式 b U a，support = Minimum support( b) ;
12. } else {
13. for each a i 在 CFP_Tree 的首部{
14. 产生 b = a i U a，support = ai. support;
15. 构造 b 的条件模式基，然后构造 b 的 CFP_Tree;
16. if CFP_Tree 不为空
17. 调用 PFP_growth ( CFP_Tree，b) ;} }
```

PFP_Growth 分析算法结合了日志自身特点，在算法分析过程中将日志中的属性进行加前缀以及属性值压缩，并设置了约束属性组，也对待分析的日志记录进行裁剪，从而提高了指定属性值日志记录的分析效率和复杂度. 该算法能够挖掘出待分析日志记录中的频繁项集，能为下一步的关联规则挖掘做准备.

3 入侵场景重构

通过 PFP_Growth 算法找出频繁项集之后，挖掘频繁项集之间关系生成关联规则. 设置支持度 Support 和置信度 Confidence，满足支持度阈值和置信度阈值的规则称为强关联规则[8-9].

关联规则的产生过程是: 设定一个最小置信度阈值 Min_Confidence，对于频繁项集 L 的非空子集 S，如果 support_count(L) /support_count(S) ≧ Min_Confidence，则有规则“S⇒L - S”.

格式化规则产生过程为: 将规则中的属性值去除，只提取出规则中的属性含义.

如何从大量的关联规则中分析出与入侵有关的规则并进行场景重构，是重点要解决的问题. 本文提出的是基于模拟攻击的格式化规则匹配方法与基于孤立点分析的场景重构方法. 规则匹配与场景重构过程如下:

(1) 实施模拟攻击，将日志进行 PFP_Growth 挖掘并生成关联规则和格式化规则;

(2) 对被攻击主机日志进行 PFP_Growth 挖掘，并产生关联规则库;

(3) 将该规则库和格式化规则进行匹配，发掘未知规则库中可疑的规则. 将规则匹配方法中格式规则库划分为相关格式化规则库和未知格式化规则库. 未知格式规则库中每个格式规则对应有一个规则链表，存储的是和该格式规则对应的详细规则.

(4) 按照入侵过程在对应的日志记录和对应的关联规则中找出详细的记录和规则;

(5) 按照可疑的属性进行孤立点分析.

(6) 根据属性相关性将对应的入侵步骤按照时间戳进行场景重构. 具体的分析步骤为: (1) 挖掘与登录主机相关的规则和记录; (2) 挖掘与漏洞扫描和缓冲区溢出相关的规则和记录; (3) 挖掘安装的代理; (4) 挖掘停止的服务; (5) 综合信息重构场景.

4 实验结果与分析

实验模拟 TCP Flood 攻击，并产生模拟攻击的日志文件，之后对日志文件进行分析，提取出和攻击步骤对应的格式化规则. 最后，分析被攻击主机的日志文件，将待匹配规则和格式化规则进行匹配，进行场景重构. 模拟的攻击过程为: 第一步，通过多次验证用户密码尝试登录被攻击主机; 第二步，在成功登录被攻击主机之后，进行端口扫描，建立连接; 第三步，安装实现攻击所需要的代理工具，并设置为开机自启动; 第四步，关闭系统中阻碍实施攻击的服务; 第五步，实施 TCP Flood 攻击.

图 2 为对防火墙日志进行挖掘，设定的约束属性组为 {“Time: [20 - 23]”，“Action: [Drop、Open]”，“P: [TCP、ICMP]”，“SIP: ”，“SP: ”，“DP: ”，“Size: ”}. 支持度为 2%，置信度为 2.5%. 挖掘出关联规则之后，通过基于模拟攻击的格式化规则匹配方法对关联规则进行匹配，匹配过程一方面可以提取与入侵相关的规则库，另一方面可以不断更新格式化规则库.

按照本文提出的场景重构方法，首先，按照需求找出和尝试登录相关的规则，并恢复出原始记录，找出可疑用户; 然后，通过防火墙相关的规则确定可疑入侵者的登录 IP; 之后，找出和扫描端口相关的规则; 再次，确认安装的代理服务器、停止的服务以及自启动项; 最后，确定入侵者的行为并按照时间先后顺序进行场景重构. 经过上面的步骤之后，得到场景重构的结果如图 3 所示.

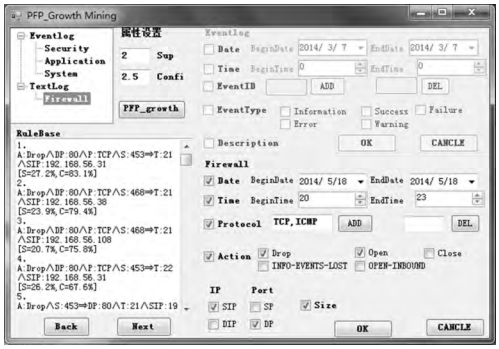

图 2 防火墙日志挖掘

图 3 场景重构结果

本文从三个方面对场景重构算法的性能进行了分析:

(1) 日志记录的数量变化对 FP_Growth 算法和 PFP_Growth 算法效率的影响;

(2) 属性约束组不同时，FP_Growth 算法和 PFP_Growth 算法的效率的比较;

(3) 重构方法的正确率.

图 4 为日志记录数量对 FP_Growth 算法和 PFP_Growth 算法挖掘效率的影响. 从图中的数据可以分析得出，日志数量的记录条数越多，PFP_Growth 算法相对 FP_Growth 算法的效率越高.

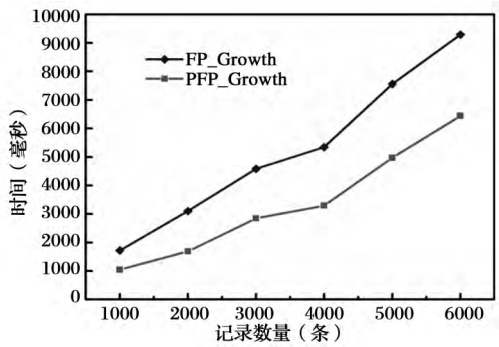

图 4 记录条数对算法的效率影响

图 5 为约束属性组设定的数量不同对 FP_Growth 算法和 PFP_Growth 算法效率的影响. 从图 5

中的数据可以分析得出，约束属性的个数设置得越少，PFP_Growth 算法的效率相对 FP_Growth 越高；约束属性的属性值设置得精确，PFP_Growth 算法的效率相对 FP_Growth 也越高.

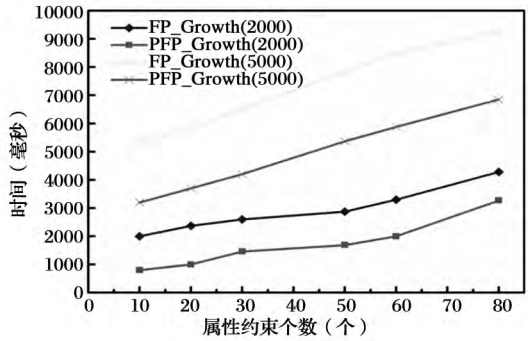

图 5　约束属性个数对算法的效率影响

图 6 是场景重构准确性评估结果. 将模拟攻击的步骤数量和成功重构出的入侵步骤数量进行了对比，从图中的数据可以分析得出，本文提出的重构方法可以重构出大部分的入侵步骤，并且准确率达到 82.3%.

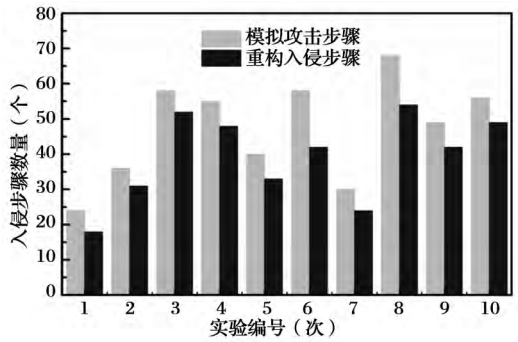

图 6　场景重构准确性评估

实验结果表明：(1) 本文提出的 PFP_Growth 算法比 FP_Growth 算法能够更高效地挖掘出与入侵相关的频繁项集；(2) 基于规则库的场景重构方法能够有效地重构出入侵行为的入侵过程，重构出的入侵场景能为证明入侵行为提供了法律上认可的有利证据.

5　结论

本文结合日志特性以及入侵行为的特征，提出了基于 FP_Growth 的日志分析算法 PFP_Growth 以及基于规则库与属性跟踪的场景重构方法. 经过实验可证明，在挖掘日志的频繁项集方面，PFP_Growth 算法比 FP_Growth 算法效率更高，在基于规则库的场景重构方法中能够更高效地重构出入侵行为过程. 据此可为入侵行为提供法律上认可的有利证据. 实验结果还表明，仅仅以日志作为电子证据进行重构取证还有局限性，难以重构出全部的入侵过程，取证分析中应该结合其它类型的证据.

参考文献

[1] Zhang W，Xu S，Zhang S. Association Rule Mining for Reasonable Curriculum Arrangement：A case study of Zhejiang University of Finance and Economics [J]. International Journal of Information Processing & Management，2015，6(2)：42－47.

[2] Zhang J，Wang S，Lv H，et al. Research on Application of FP－growth Algorithm for Lottery Analysis [M]//LISS 2013. Springer Berlin Heidelberg，2015：1227－1231.

[3] Shrivastava N，Khanna R. FP－Growth Tree Based Algorithms Analysis：CP－Tree and K Map [J]. Binary Journal of Data Mining & Networking，2015，5(1)：26－29.

[4] 张辰，孟少卿，鹿凯宁. 基于数据挖掘和蜜罐的新型入侵检测系统研究 [J]. 电子设计工程，2012，20(14)：109－112.

[5] Zeng Y，Yin S，Liu J，et al. Research of Improved FP－Growth Algorithm in Association Rules Mining [J]. Scientific Programming，2015：1－6

[6] Sidhu S，Meena U K，Nawani A，et al. FP Growth Algorithm Implementation [J]. International Journal of Computer Applications，2014，93(8)：6－9.

[7] 王玲，钱华林. 计算机取证技术及其发展趋势 [J]. 软件学报. 2003，14(9)：1635－1644

[8] 张辰，孟少卿，鹿凯宁. 基于数据挖掘和蜜罐的新型入侵检测系统研究 [J]. 电子设计工程，2012，20(14)：109－112.

[9] 杨云，罗艳霞. FP－Growth 算法的改进 [J]. 计算机工程与设计，2010，31(7)：1506－1509.

[10] 葛贤银，韦素媛. 一种基于 BM 算法的改进模式匹配算法研究 [J]. 计算机应用技术，2009，32(20)：73－75.

责任编辑　徐　刚

Research on Forensics Analysis Methods Based on Windows Logs

SHEN Yue-li

(Department of Computer Engineering, Taiyuan Institute of Technology, Taiyuan 030008, China)

Abstract: A scheme based on scene reconstruction of host log is realized. Associated with a redundant data cleaning technology based on EventID classification model, a FP_Growth-dependent log analysis algorithm PFP_Growth, a formatting rules matching method based on simulation attack, as well as a scene reconstruction avenue based on the rule database and attributes tacking are proposed. Additionally, on the one hand, the efficiency of both the PFP_Growth and the FP_Growth algorithm are compared simultaneously, validating the high efficiency of the former; on the other hand, the effectiveness of the proposed scheme is verified experimentally. The reconstructed invasion scenes proved evidences for the intrusion.

Key words: computer forensics; Windows logs; PFP_Growth; matching formatting rules; scene reconstruction

(上接第 61 页)

The Human-Machine Collaboration Translation Method Based on English-Chinese Parallel Military Corpus

HUANG Jin-zhu[1], FAN Xin-zhan[2], LI Feng[3,4], ZHANG Ke-liang[1]

(1. Department of Language Engineering, PLA University of Foreign Languages, Luoyang 471003, China;
2. PLA 69018 Unit, Kashi 844000, China;
3. Logistics Science Research Institute of PLA, Beijing 100166, China;
4. School of Computer Science and Engineering, Beihang University, Beijing 100191, China)

Abstract: The paper, based on actual requirement of military literature translation, puts forward the initiative of an English-Chinese parallel corpus construction with large amount of open-source corpora. The paper also puts forward a human-machine Collaboration method of processing military texts based on the self-constructed parallel military corpus and Trados CAT system. Through comparative analysis, the efficiency of the method is tested. Besides, in order to solve the problems encountered in the process of constructing the corpus and later corpus updating, the paper puts forward a sentence alignment method based on lexical meaning. Through comparative analysis, the method is also proved to be effective in solving the problem of sentence alignment in processing military text.

Key words: parallel military corpus; India; Trados CAT system; sentence alignment; human-machine collaboration