ARS：基于文件行为的勒索软件主动防御技术研究

田　锋，周安民，刘　亮，张　磊
(四川大学网络空间安全学院，成都 610065)

摘　要：针对勒索软件(Ransomware)造成的安全威胁问题，本文在分析多种勒索软件家族攻击特点和传统恶意代码检测技术的基础上，提出一种基于文件行为的勒索软件主动防御技术(Anti-Ransomware System，ARS)．该技术采用基于文件行为统计异常的方法进行勒索软件检测，通过文件过滤驱动收集正常软件与勒索软件动态文件行为信息用作训练集，使用多种分类算法训练并生成勒索软件检测分类器，用于运行时检测勒索软件．利用写时复制技术动态备份程序运行时修改的文件，根据检测结果决定是否恢复文件．最后，设计实现原型系统并进行了测试．实验结果表明，在确保数据文件安全性的前提下，ARS 能够有效地防御勒索软件，减小勒索软件攻击的危害．

关键词：勒索软件；文件行为；主动防御；分类算法；文件过滤驱动
中图分类号：TP391.1　　**文献标识码**：A　　DOI：10.19907/j.0490-6756.2021.023001

ARS：research on proactive defense technology of ransomware based on file behavior

TIAN Feng，*ZHOU An-Min*，*LIU Liang*，*ZHANG Lei*
(College of Cybersecurity，Sichuan University，Chengdu 610065，China)

Abstract：In order to reduce the loss caused by the attack of ransomware，this paper proposed a framework (Anti-Ransomware System，ARS) based on the family characteristics and the traditional malicious code detection technology. Specifically，a method is proposed to detect ransomware based on file behavior statistics. It uses the minifilter to collect the file behavior information of benign and ransomware as train set，and uses a variety of supervised classification algorithms to train classifier for run-time detection. Then，the copy-on-write technology is utilized to dynamically backup modified files in the program at runtime，and the detection results are used to determine whether to restore files. Finally，a prototype system is developed and tested. The results show that，under the premise of ensuring the security of the data file，the ARS framework can effectively prevent the ransomware attack and reduce the harm of ransomware.

Keywords：Ransomware；File behavior；Proactive defense；Classification；Minifilter

收稿日期：2020-07-14
基金项目：国家重点研发计划(2017YFB0802900)
作者简介：田锋(1995—)，男，四川巴中人，硕士研究生，研究方向为恶意代码检测．
通讯作者：张磊．E-mail：zhanglei2018@scu.edu.cn

1 引　言

勒索软件是一种通过劫持受害者数字资产后索要赎金以牟取利益的恶意代码[1]. 勒索软件攻击会对受害者的数字资产安全造成严重危害,并且这类攻击呈现一种上升趋势[2],急需研究有效的防御措施.

针对勒索软件攻击,Brewer 将勒索软件攻击划分为 5 个阶段:传播感染、执行、破坏备份、执行加密和勒索,并预测未来勒索软件攻击将会更普遍,且更具破坏性[3]. Kharraz 等通过对 2006 年到 2014 年出现的 1 359 个勒索软件样本分析发现,勒索软件与正常软件访问文件系统的请求方式有显著的不同,可以通过监控一些敏感函数及文件系统检测勒索软件[4]. Luo 等提出通过规范企业安全管理策略及提高使用者安全意识能减少勒索软件攻击,但并未从技术层面提出解决方案[5]. Andronio 设计的 Heldroid 系统在安卓平台上实现勒索软件检测,通过监控有关加密的系统调用及勒索信息检测勒索软件[6],勒索信息提取自静态资源及运行时网络流量中的文本信息,对这些文本信息进行自然语言处理,通过机器学习提取勒索信息特征,利用这些特征来检测勒索软件,但在桌面系统这些措施很容易被绕过. Cabaj 等对 CryptoWall 和 Locky 勒索软件家族的样本 HTTP 数据包分析后,提出使用 HTTP 流量特征检测有网络行为的加密类勒索软件[7],设计出 SDN 勒索软件快速检测系统,但并未证明该方法是否能有效检测未知勒索软件. Moore 提出利用蜜罐技术快速检测勒索软件,通过监控诱饵文件判定程序是否为勒索软件,但该方法会存在较高的漏报率[8]. 殷明等提出固态硬盘的防范技术,利用固态存储设备数据非原位更新、修改垃圾回收等策略进行数据备份[9]. Lee 等设计实现基于云平台的 CloudRPS 勒索软件防御系统,通过实时监控网络、文件和服务器等信息检测勒索软件,能够减少因为勒索软件攻击造成的损失[10]. Kim 等设计一种结合社会工程学的勒索软件检测系统,能够分析其静态和动态特征[11]. Kharraz 等在已有研究基础上,通过监控文件系统活动及屏幕的变化,利用文本分析技术快速检测勒索软件,设计了 UNVEIL 自动检测系统,实验证明该系统有较高的准确率及较低的假阳性率[12]. 赵翠镕等提出基于语义 API 依赖图的真机动态分析方法,运用集成学习-随机深林进行恶意代码分类,该方法同样可用于勒索软件检测[13]. Scaife 等对 14 个家族 496 个勒索软件样本进行分析,总结出勒索软件家族基因是其加密用户文件这一行为,设计以用户数据为中心的勒索软件主动防御系统 CryptoDrop,但该系统会造成部分文件丢失[14]. 然而,上述方法基于特征码的方式更新速度相对较慢,能通过加壳、混淆等方式绕过;启发式扫描会因为勒索软件不修改注册表,也没有注入等行为,不适应于勒索软件检测[15];沙箱的系统开销较大,监控加密函数不能发现使用自有加密算法的勒索软件,便会导致误报.

与上述工作不同,本文提出一种基于文件行为的勒索软件主动防御技术——ARS,从写时拷贝技术和恶意代码分类得到启发,通过分类算法在运行时判定程序是否为勒索软件,利用文件过滤驱动技术备份程序运行过程中修改的文件,既能有效地检测勒索软件也能防止勒索软件运行所带来的危害. 提出的反勒索框架不依赖勒索软件静态特征;也无需部署蜜罐系统来捕捉勒索软件行为;同时不通过算法逆向的方式来恢复被加密的文件. 利用提取的勒索软件运行时特征,能够在勒索软件运行初期阻止其继续执行,使用户文件得到保护. 主要工作和贡献包括以下几方面.

(1) 提出一种基于运行时文件行为的勒索软件主动防御框架. 针对勒索软件检测问题,在现有恶意代码分类方法的基础上,受(Copy-On-Write, COW)技术的启发,提出一种勒索软件运行时分类方法的检测框架. 该系统可以根据勒索软件文件行为特征生成勒索软件通用检测规则,并根据新样本动态更新行为规则模型.

(2) 在未知勒索软件检测方面,ARS 通过文件行为特征生成检测规则,在程序运行时通过同样的方式提取文件行为特征与之前的规则进行匹配,采用多算法 Bagging 方式将支持向量机 SVM,决策树 DT,朴素贝叶斯 NB 以及随机森林 RF 算法作为候选算法集合进行多数投票,以此判断该程序是否为勒索软件.

(3) 设计实现 ARS 原型系统并进行测试评估,利用文件过滤驱动备份程序在运行时修改的文件,能够及时回滚勒索软件对文件的操作,根据检测的结果判断是否还原文件,即使检测结果出现漏报也不会造成文件丢失;不会影响系统与用户正常的文件操作,降低了因为勒索软件攻击所造成的损失;另外,在勒索软件运行阶段检测,仅包含几个

Windows 内核模块，不需要在用户网络中添加其它设备且与原有的杀毒软件共存，同时框架各模块具备良好的扩展性. 测试结果表明，ARS 能够有效检测勒索软件，并恢复被修改的文件且对系统效率影响较小.

2　基于文件行为的勒索软件主动防御框架

2.1　概　述

由于勒索软件与正常软件在文件行为特征上存在差异，ARS 通过文件过滤和驱动来收集行为特征，并对特征熵进行归一化处理，从而实现分类器的输入. 训练采用贝叶斯分类、决策树、支持向量机和随机森林等多种算法进行特征学习并生成勒索软件检测分类器. 利用主动防御的方式检测勒索软件，即勒索软件在加密文件之前被检测到，且在勒索软件执行过程中使用 COW 技术保护文件.

ARS 主动防御框架由三部分组成：勒索软件行为规则学习模块、主动防御模块和文件保护模块. 总体技术框架如图 1 所示.

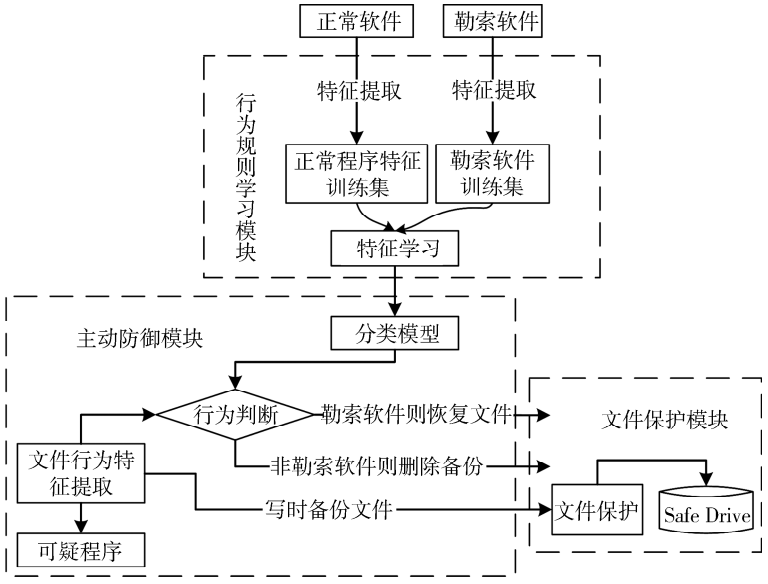


图 1　基于文件行为的勒索软件主动防御框架

Fig. 1　Ransomware active defense framework based on file behavior

在图 1 中，勒索软件行为规则学习模块根据勒索软件与正常软件的文件行为特征生成勒索软件检测分类器. 通过文件过滤驱动收集勒索软件样本与正常程序的动态文件行为 IRP 记录作为训练集，经过分析后提取文件行为相关信息，利用多种监督机器学习算法训练生成检测分类器.

文件保护模块负责文件备份及还原功能，主动防御模块负责运行时监控并记录程序文件行为，利用行为规则学习模块生成的分类模型动态判定程序行为，终止勒索软件运行，并依据程序行为特征与文件保护模块进行交互，根据当前程序是否为勒索软件决定是否恢复备份文件.

2.2　勒索软件行为规则学习

2.2.1　特征记录　通过 ARS 中的文件过滤驱动模块收集勒索软件与正常软件的文件特征，过滤驱动通过在系统中注册文件行为操作回调函数，在回调函数中通过解析文件操作 IRP 请求包，记录文件操作请求行为. 具体文件操作记录项如表 1 所示.

表 1　文件操作特征项

Tab. 1　File operation feature item

文件行为特征	说明
Time	文件操作时间
Pid	当前进程号
Name	当前进程名称
Operation	文件行为操作类型
Entropy	文件写入熵值
File	文件路径

在表 1 中，进程号和进程名称用于确定当前文件操作的程序，文件操作类型通过 IRP 请求包中的 Major Function 及 Minor Function 表示.

经过加密的数据流一般和完全随机的数据流相似，即被加密的数据比原始数据表现出更大的不确定性，用信息熵表示值就更大[16]. 假设每一个写入缓冲文件块中的字节成均匀分布，则写入熵即香

农信息熵计算公式如式(1)所示.

$$H(d) = -\sum_{i=1}^{n} p(i)\log_2 p(i) \tag{1}$$

其中,d 表示一个文件 I/O 数据块随机事件;$p(i)$ 表示 d 中的 n 个字符中,其中第 i 个出现的概率. 取 8 位计算一次,总共有 256 种可能,则熵值取值范围为 0 到 8 之间,为计算方便,最后的平均熵值结果除以 8 进行归一化处理.

2.2.2　特征处理方法　在勒索软件攻击执行的过程中,需要进行大量文件操作,如遍历文件夹、文件读写,以及写入高熵值加密数据等操作[17],通过对勒索软件与正常软件原始文件行为操作数据进行分析后,本文基于勒索软件文件行为统计异常特征选择检测特征,设计检测方法. 其中,正常软件与勒索软件写入熵值数据核心密度估计(Kernel Density Estimate, KDE)如图 2 所示,而勒索软件与正常软件行为特征核心密度估计如图 3 所示.

由图 2 和图 3(a)~(d)可以发现勒索软件与正常软件的文件行为存在很大的不同,正常软件的文件操作行为基本都在一个接近 0 的水平,而勒索软件的文件操作行为在各个区间都有,基本都高于正常软件,因此,利用这个区别和特点,可以基于文件行为的构建勒索软件分析模型.

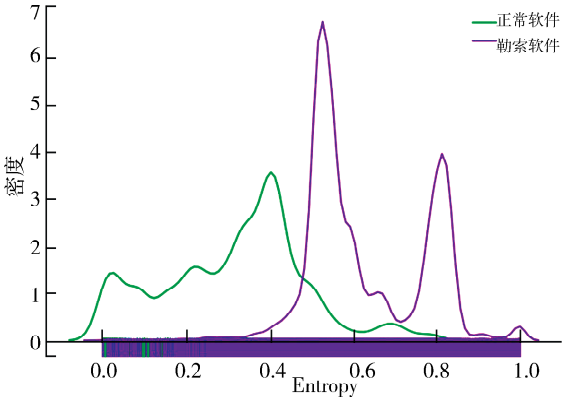
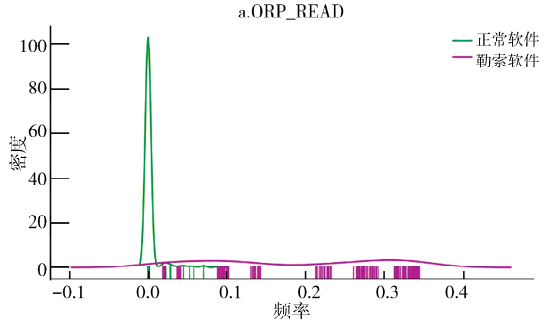
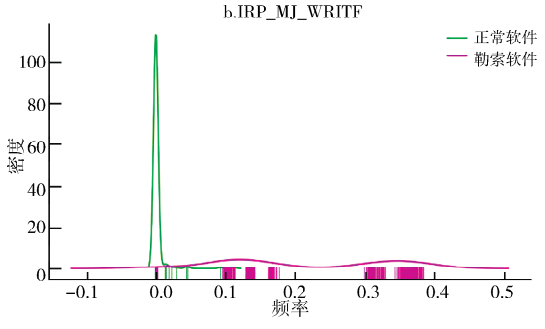


图 2　正常软件与勒索软件写入熵值数据核心密度图

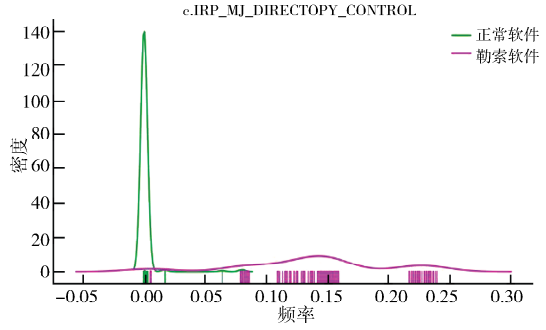
Fig. 2　Normal software and ransomware write entropy data core density map



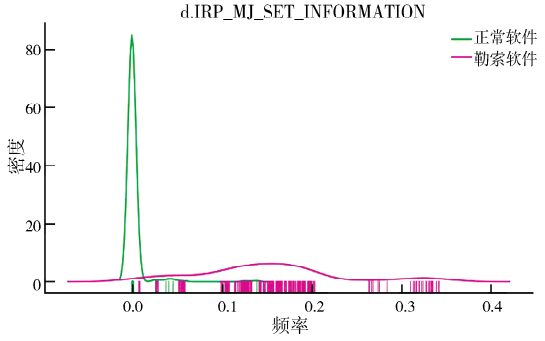
(a) 文件读时核心密度



(b) 文件写时核心密度



(c)文件目录控制核心密度



(d)文件信息设置核心密度

图 3　勒索软件与正常软件行为特征核心密度图

Fig. 3　Core density map of ransomware and normal software behavior characteristics

勒索软件要加密用户文件必然有相关文件行为特征,本文主要选取 5 个文件行为特征,包括:读、写. 用于基于文件行为的勒索软件检测,最后得到基于文件行为的特征集:File_Op_Feature={R, W, D, S, E},各个特征具体含义如表 2 所示.

大部分勒索软件在执行加密用户文件的过程中,会在短时间内有大量文件操作行为. 如果使用基于单位时间的文件行为进行勒索软件行为归一化,会因为勒索软件加密算法、计算机性能等因素,导致各个文件行为特征在不同计算机上不同,同一时间内的特征频率相差很大,最终会影响训练生成的分类器的分类效果. 因此,本文采用基于主机现

有文件及文件夹总数的归一化方法，虽然文件或文件夹总数在不同主机也不相同，但是基于文件或文件总数的归一化方法能够反应勒索软件加密用户文件的趋势，将数据统一在同一个量纲区间，消除特征量纲的影响，使特征均在同一个参考系下. 使用正确归一化方法处理后的特征训练的分类器能够更快收敛，加快梯度下降求解最优解的速度并且有可能提高分类精度.

表 2　文件操作特征项

Tab. 2　File action feature item

特征	归一化方法	说明
R->IRP_MJ_READ	读文件频率，针对单个文件的 IRP_MJ_READ 只计一个，使用系统总文件数归一化	勒索软件要加密文件必须首先读文件
W->IRP_MJ_WRITE	写文件频率，多个只计一个，使用系统总文件数归一化	加密用户文件必然伴随写文件操作
D->IRP_MJ_DIRECTORY_CONTROL	文件夹遍历频率，多个只计一个，使用系统总文件夹数归一化	加密文件过程中，勒索软件需要遍历夹，寻找目标文件
S->IRP_MJ_SET_INFROMATION	文件重命名/删除等改变频率，多个只计一个，使用系统总文件数归一化	勒索软件加密用户文件后一般会修改文件后缀或者删除原文件
E->Write Entrony	文件写入熵值	写入的加密数据类似随机数据，熵值较高

2.2.3　检测算法　恶意代码检测领域的分类问题一般都是二分类问题[18]，目前已有很多分类算法被成功应用到恶意代码检测领域，常用的分类算法包括贝叶斯分类、决策树、支持向量机和随机森林等[19].

本文选用多算法 Bagging 的方式，因为它能增强对小数据集的处理能力，对噪声或异常值的敏感性较低，而且并行结构的方式便于高效实现（如算法 1 所示），算法 1 将支持向量机 SVM，决策树 DT，朴素贝叶斯 NB 以及随机森林 RF 算法作为候选算法集合. 将收集的勒索软件特征数据集 D 抽取出 T 个含有 n 个训练样本的集合. 在每个训练集上训练出分类器. 在对预测输入进行整合时，本文选用多数投票（Majority vote）方式，对分类器进行选择.

2.3　文件保护

文件保护模块主要完成文件写时备份与文件恢复功能，受写时拷贝技术启发[20-21]，当发现有新的以写或删除句柄的方式打开文件时，文件保护系统会将该文件拷贝至安全存储空间 SafeDrive，SafeDrive 通过建立信任程序的方式拒绝其它非受信进程文件操作请求来保证只读存储空间的安全性，SafeDrive 可以是本地硬盘空间也可以是移动磁盘空间，用户可以自由设置，文件保护流程如图 4 所示.

算法 1　Bagging 算法

输入　勒索软件训练集 $D=\{x_1, x_2, \cdots, x_m\}$；
基学习算法集 $L=\{SVM, DT, NB, RF\}$；
训练轮数 N；

输出　最佳算法 $R(x)$

1) for $L \in \{SVM, DT, NB, RF\}$ do
2)　　for $n=1,2,\cdots,N$ do
3)　　$h_n = L(D, D_{bs})$；
4)　　end for
5)　　$H(x) = \underset{y \in \gamma}{\operatorname{argmax}} \sum_{n=1}^{N} \operatorname{diff}(h_n(x) = y)$
6)　　add $H(x)$ to R
7) end for
8) $R(x) = \operatorname{argmax} \sum_{t=1}^{4} \operatorname{diff}(R_t)$
9) end

通过 FltRegisterFilter 注册与 IRP_MJ_CREATE 相关联的后操作回调函数，在后操作回调函数中通过解析 PFLT_CALLBACK_DATA 结构体中的 Iopb 成员属性判断当前文件操作是否为以写或删除权限打开文件，具体判断条件为

FlagOn(
Data->Iopb->Parameters. Create. SecurityContext->DesiredAccess,
FILE_WRITE_DATA | FILE_APPEND_DATA |
DELETE | FILE_WRITE_ATTRIBUTES | FILE_WRITE_EA |
WRITE_DAC | WRITE_OWNER | ACCESS_SYSTEM_SECURITY)

由于在备份文件的过程中也需要使用 FltCreateFile 打开文件，与应用层打开文件一样也会产生文件操作 IRP 请求，如果不过滤这种文件操作 IRP 请求会循环嵌套造成死循环. 因此，通过宏 FLT_IS_FS_FILTER_OPERATION 判断是否为过滤驱动发送的数据，如果为过滤驱动数据则直接

返回 FLT_PREOP_SUCCESS_NO_CALLBACK，不对该 IRP 请求进行处理同时也不会调用该过滤驱动的后操作回调函数.

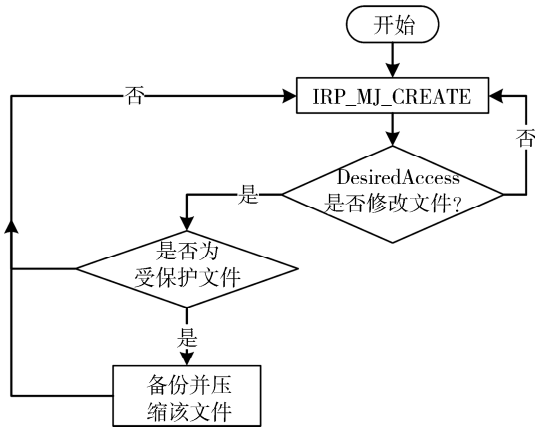


图 4　文件保护工作流程
Fig. 4　File protection workflow

2.4　主动防御

主动防御模块主要负责文件保护模块与行为规则学习模块之间的联动，通过读取行为学习模块生成的检测规则判断程序是否为勒索软件，然后决定是否还原备份文件，各个模块紧密耦合联动，主动防御模块与其它模块交互，如图 5 所示. 主要功能为文件监控、勒索软件检测与清除，模块具体实现分为应用层与内核层驱动模块，应用层负责根据文件行为规则检测程序是否为勒索软件，内核层负责文件行为监控与清除勒索软件.

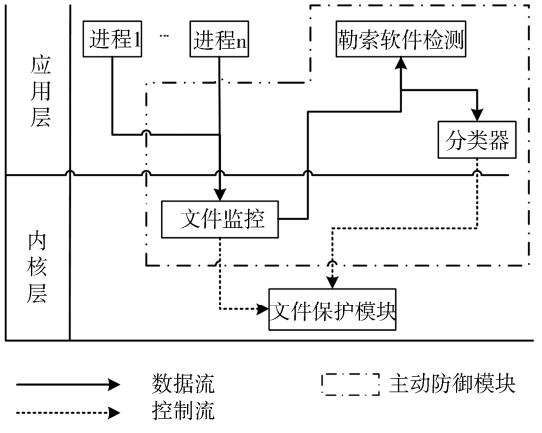


图 5　主动防御与其他模块的关系
Fig. 5　Proactive defense against other modules

为减小因监控文件操作对系统效率的影响，对于耗时的操作，如计算写入信息熵利用单独内核线程异步计算，记录完成后直接将记录结果发往应用层统一处理. 归一化因子主要包括当前系统文件及文件夹总数，在文件监控模块启动时通过单独内核线程遍历当前磁盘，获得当前系统文件及文件夹总数量. 为防止勒索软件通过创建大量文件及文件夹的方式影响分类特征，归一化因子非实时更新，采用定期更新的方式，间隔时间为 30 min.

应用层在接收到内核层发送的程序文件操作信息后根据进程名及进程号，对各个进程的文件操作进行特征处理.

3　测试评估与分析

搭建测试环境时，勒索软件文件行为获取环境基于 Cuckoo 沙盒及文件记录模块搭建而成[22]，沙盒会模拟用户操作鼠标等. 虚拟机中的操作系统为 Windows 7(x64)，系统中初始安装常用软件，如浏览器、办公软件等并放入常用文件如 doc、jpg 等，环境尽量接近真实使用的计算机，每次执行 60 min，执行完后取出文件行为日志并恢复虚拟机快照.

3.1　数据集

本文收集到 542 个正常程序和 6 种勒索软件家族共计 244 个勒索软件样本文件行为特征，勒索软件本主要来自 VirusShare、Malwr 及相关安全论坛收集，最终的 244 个样本均为存活的勒索软件样本，勒索软件样本分布及特征记录情况如表 3 所示.

表 3　勒索软件样本分布及特征记录
Tab. 3　Ransomware sample distribution and feature record

勒索软件家族	样本数量	大小/MB	IRP 数量/万条
Locky	54	734	5128
TeslaCrypt(Tescrypt)	48	326	2248
Cerber	33	453	3038
XTBL	11	217	1620
CryptXXX	13	154	1056
Crowti(CryptoWall)	85	643	4693

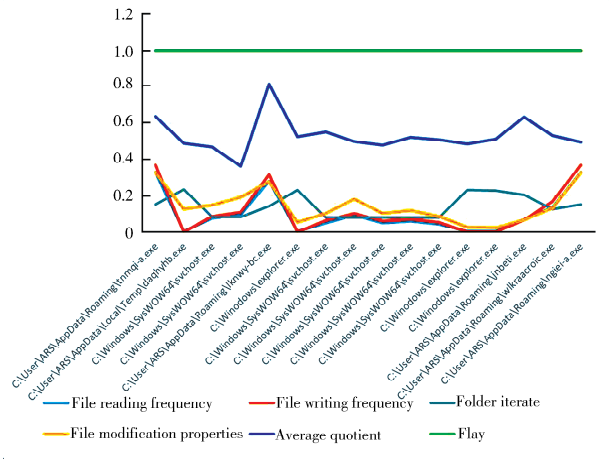


图 6　文件行为特征记录
Fig. 6　Document behavior record

3.2　分类检测评估

以进程为单位提取其文件行为 5 元组特征，包括文件写频率、文件读频率、文件夹遍历频率、修改文件属性频率及文件平均写入熵值，最终经过处理的文件行为 5 元组{R，W，D，S，E}的特征格式如图 6 所示，最后一位为标识位，1 表示勒索软件进程，0 表示正常软件进程.

表 4　检测实验结果

Tab. 4　Experiment results

分类方法	TPR/%	FPR/%	准确率/%
支持向量机 SVC(kernel='rbf')	97.8	2.9	97.6
决策树 DecisionTreeClassifier	97.8	1.7	97.9
朴素贝叶斯 MultinomialNB	96.9	2.1	97.2
随机森林 n_estimators=100	99.4	2.4	98.9
算法集成 Bagging	98.5	1.2	98.6

因为提取到的文件行为特征集较小，所以未选择对样本集要求较大的分类算法，如 k 邻近算法等. 对提取到的特征集合分别采用支持向量机、决策树、朴素贝叶斯及随机森林方法进行训练，采用 10 折交叉验证对基于文件行为统计异常的训练生成的分类器进行验证，设勒索软件为阳性，正常软件为阴性，最终验证结果如表 4 所示.

使用 sklearn. metrics 提供的 roc_curve 方法，做出 10 折交叉验证 ROC 曲线如图 7 所示.

通过表 4 的分类效果及图 7 的 ROC 曲线，表明测试所使用的分类算法对特征都具有较好的分类效果，准确率均达到 0.96 以上且 FPR 均控制在 0.03 以下. 支持向量机与决策树取得了相同的 TPR 值，但是其 FPR 值达到了 0.029. 从实验结果可知，朴素贝叶斯效果最不理想. 这是由于各个特征存在一定的相关性，而朴素贝叶斯算法假定各个特征独立不相关，模型较为简单，不能学习各个文件特征之间的相关性，所以有较低的 TPR 和较高的 FPR 值，而在随机森林算法中，这个问题得到一定缓解. 在准确率和 TPR 值上，随机森林有着最好的结果，但是 Bagging 集成算法有着更低的 FPR 为 1.2%. 在恶意代码检测实践中，更低的 FPR 意味着更少的漏报. 由于与勒索软件对抗过程中，少量的漏报也能带来巨大的数据损失，从而在实际应用中算法集成方法具有减少风险的良好效果.

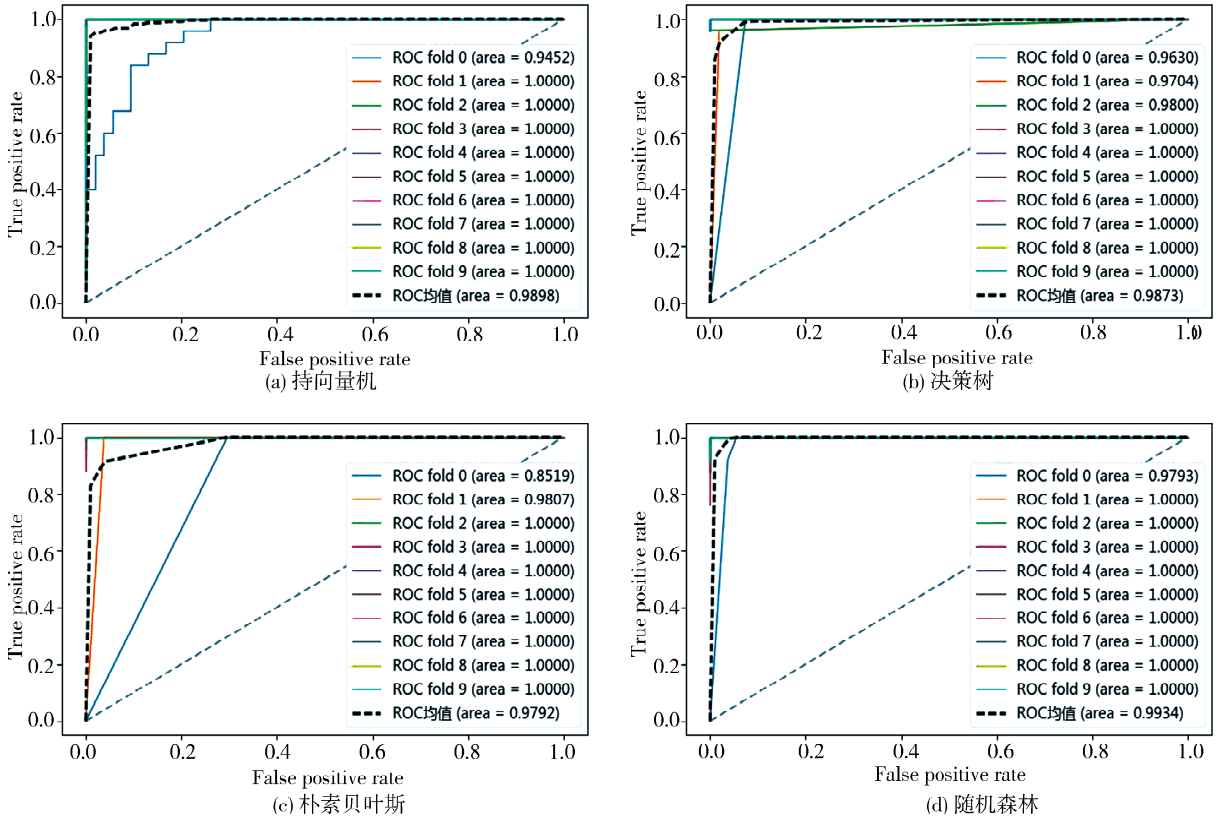


图 7　检测分类结果 10 折 ROC 曲线

Fig. 7　10-fold ROC curve of detection classification results

在 10 折 ROC 曲线图中,所有分类算法的 AUC 均值都在 0.97 以上,与表 4 中反应的结果一样,所有算法都有不错的效果. 但是不同算法的 10 折 ROC 曲线的变化不同,由于随机森林算法为集成算法,内部采用多颗决策树进行分类学习. 最后,采用所有分类结果最多的类作为最后的结果,算法容错能力较好,因此分类结果波动较小. 而其他算法均为非集成分类算法,分类容错能力较差,而造成这个问题的根本原因是由于勒索软件样本中存在一些有注入行为的样本,导致以进程为单位进行筛选勒索软件文件行为特征时出现错误,使最终训练集合中出现了噪声数据,影响了其他分类算法的分类效果.

3.3　文件保护测试

在实验 1 进行勒索软件检测测试的同时,同步测试文件备份还原功能,在测试系统中部署 783 个勒索软件常用攻击文件并保存初始 md5 值,在测试后还原被加密的文件,最后经过分析所有文件都被正确还原,ARS 保护文件保护文件能力达到 100%. 然后对文件保护功能进行安全性测试,保证备份空间不被其它应用程序访问、修改及删除操作,测试是将 C:\SafeDrive 设置为备份空间,测试过程和结果如表 5 所示.

表 5　文件保护攻击测试

Tab. 5　File protection attack test

测试内容	测试方法	预期结果	测试结果
文件访问	直接打开文件夹	访问失败	成功
文件删除	命令行 del /F /S /Q C:\SafeDrive	删除失败	成功
文件修改	重命名 SafeDrive 文件夹	修改失败	成功
文件添加	命令行 copy 文件到 SafeDrive	添加失败	成功

3.4　系统性能测试

ARS 系统主要通过在文件系统驱动层上添加一层过滤驱动层实现监控文件操作,所有的文件操作 IRP 均要通过文件过滤驱动,因此效率测试实验主要通过在安装 ARS 原型系统及未安装 ARS 原型系统进行相关文件操作,最后对比所需要的时间,对比安装 ARS 后对系统文件操作效率的影响,具体实验及实验结果如图 8 所示.

在图 8 中,安装 ARS 原型系统后,由于会对系统文件操作进行监控,相比原系统有一定效率损失,但是因为对耗时操作均使用单独的内核线程执行,因此对系统总体效率影响不大,全部测试文件操作时间增加 7.9%,但是分散到每一个文件的带来的效率损失几乎可以忽略不计,能保证加载 ARS 模块后的系统可用性.

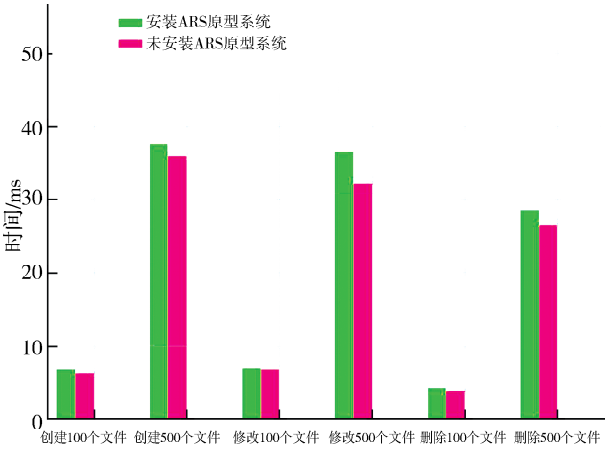


图 8　ARS 原型系统对文件操作效率测试

Fig. 8　ARS System operation efficiency test

4　结　论

本文针对勒索软件的危害和防御检测存在的问题,提出一种基于文件行为的勒索软件主动防御框架 ARS,在运行时利用监控到的程序文件行为检测勒索软件并对其修改的文件进行备份保护,降低因为勒索软件攻击所造成的损失. 根据设计 ARS 勒索软件主动防御框架开发原型系统并进行相关测试,测试结果表明,ARS 能够有效检测勒索软件,恢复被修改的文件,并且对系统效率影响较小. 在未来的工作中将对框架改进以适用对文件的缓慢攻击类的勒索软件检测,同时增加区分勒索软件和正常加密软件的机制.

参考文献:

[1] Everett C. Ransomware: To pay or not to pay? [J]. Comput Fraud Secur, 2016(4): 8.

[2] Trend Micro XGen™. Ransomware repercussions: baltimore county sewercharges, 2 medical services temporarily suspended. accessed[EB/OL]. (2019-07-04)[2020-03-22]. https://www.trendmicro.com/vinfo/us/securty/news/cybercrime-and-digital-threats/ransomwarerepercussions-baltmore-county-sewer-charges-2-medical-servicestemporarily-suspended.

[3] Brewer R. Ransomware attacks: detection, prevention and cure [J]. Netw Secur, 2016(9): 5.

[4] Kharraz A, Robertson W, Balzarotti D, *et al*. Cut-

ting the gordian knot：A look under the hood of ransomware attacks[C]//Proceedings of the International Conference on Detection of Intrusions and Malware，and Vulnerability Assessment. Cham：Springer，2015.

[5] Luo X，Liao Q. Awareness education as the key to ransomware prevention[J]. Inf Syst Secur，2007，16：195.

[6] Andronio N. Heldroid：Dissecting and detecting mobile ransomware[C] // Proceedings of the 18th international conference on Research in Attacks，Intrusions，and Defenses. Kyoto：Springer，2015.

[7] Cabaj K，Gregorczyk M，Mazurczyk W. Software-defined networking-based crypto ransomware detection using HTTP traffic characteristics [J]. Comput Electr Eng，2016，66：353.

[8] Moore C. Detecting ransomware with honeypot techniques[C] //Proceedings of the Cybersecurity and Cyberforensics Conference. Amman：IEEE，2016.

[9] 殷明，贾世杰. 一种局域网中基于 SSD 的防范勒索软件攻击技术[J]. 信息网络安全，2019，19：71.

[10] Lee J K，Moon S Y，Park J H. CloudRPS：a cloud analysis based enhanced ransomware prevention system [J]. J Supercomput，2017，73：3065.

[11] Kim D，Kim S. Design of quantification model for ransom ware prevent [J]. World J Eng Tech，2015 3：203.

[12] Kharraz A，Arshad S，Mulliner C，*et al*. UNVEIL：a large-scale，automated approach to detecting ransomware[C] //Proceedings of the 25th USENIX Security Symposium，Austin：USENIX. 2016.

[13] 赵翠镕，方勇，刘亮，等. 基于语义 API 依赖图的恶意代码检测 [J]. 四川大学学报：自然科学版，2020，57：488.

[14] Scaife N，Carter H，Traynor P，*et al*. Cryptolock (and drop it)：stopping ransomware attacks on user data [C]//Proceedings of the 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS). Nara，Japan：IEEE，2016.

[15] Continella A，Guagnelli A，Zingaro G，*et al*. ShieldFS：a self-healing，ransomware-aware filesystem [C]//Proceedings of the 32nd Annual Conference on Computer Security Applications. New York：ACM SIGCOMM，2016.

[16] Guo X，Liu R，Li P，*et al*. Enhancing extractable quantum entropy in vacuum-based quantum random number generator [J]. Entropy，2018，20：819.

[17] Moussaileb R，Bouget B，Palisse A，*et al*. Ransomware's early mitigation mechanisms[C]//Proceedings of the 13th International Conference on Availability，Reliability and Security. Hamburg，Germany：ACM，2018.

[18] Zhang H，Xiao X，Mercaldo F，*et al*. Classification of ransomware families with machine learning based on N-gram of op-codes [J]. Future Gener Comp Sy，2019，90：211.

[19] Sharmeen S，Ahmed Y A，Huda S，*et al*. Avoiding future digital extortion through robust protection against ransomware threats using deep learning based adaptive approaches [J]. IEEE Access，2020，8：24522.

[20] Zhang H，Xiao X，Mercaldo F，*et al*. Classification of ransomware families with machine learning based on N-gram of opcodes [J]. Future Gener Comp Sy，2019，90：211.

[21] Jin Y，Tomoishi M，Matsuura S，*et al*. A secure container based backup mechanism to survive destructive ransomware attacks[C]//Proceedings of the International Conference on Computing，Networking and Communications. Maui，HI，USA：IEEE，2018.

[22] Sandbox C. Cuckoo sandbox-automated malware analysis [EB/OL]. (2019-11-20)[2020-03-25]. https：//cuckoosandbox. org/.

引用本文格式：

中　文：田锋，周安民，刘亮，等. ARS：基于文件行为的勒索软件主动防御技术研究[J]. 四川大学学报：自然科学版，2021，58：023001.

英　文：Tian F，Zhou A M，Liu L，*et al*. ARS：research on proactive defense technology of ransomware based on file behavior [J]. J Sichuan Univ：Nat Sci Ed，2021，58：023001.