DOI:10.7544/issn1000-1239.2019.20190394
56(10):2262-2276,2019

一种基于软件定义安全和云取证趋势分析的云取证方法

刘雪花[1,2] 丁丽萍[1,3,4] 刘文懋[5] 郑 涛[6] 李彦峰[1,2] 吴敬征[7]

[1](中国科学院软件研究所并行软件与计算科学实验室 北京 100190)

[2](中国科学院大学计算机科学与技术学院 北京 100049)

[3](广州中国科学院软件应用技术研究所电子数据取证实验室 广州 511458)

[4](广东中科实数科技有限公司 广州 511458)

[5](北京神州绿盟信息安全科技股份有限公司 北京 100089)

[6](中国联合网络通信有限公司 北京 100033)

[7](中国科学院软件研究所智能软件研究中心 北京 100190)

(xuehuagao@qq.com)

A Cloud Forensics Method Based on SDS and Cloud Forensics Trend Analysis

Liu Xuehua[1,2], Ding Liping[1,3,4], Liu Wenmao[5], Zheng Tao[6], Li Yanfeng[1,2] and Wu Jingzheng[7]

[1](*Laboratory of Parallel Software and Computational Science, Institute of Software, Chinese Academy of Sciences, Beijing* 100190)

[2](*School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing* 100049)

[3](*Digital Forensics Laboratory, Institute of Software Application Technology, Guangzhou and Chinese Academy of Sciences, Guangzhou* 511458)

[4](*Guangdong Chinese Academy of Sciences & Realdata Science and Technology Company Limited, Guangzhou* 511458)

[5](*NSFOCUS Information Technology Company Limited, Beijing* 100089)

[6](*China United Network Communications Corporation Limited, Beijing* 100033)

[7](*Intelligent Software Research Center, Institute of Software, Chinese Academy of Sciences, Beijing* 100190)

Abstract With the development and popularization of cloud computing, the security situation of cloud computing environment is getting worse. Cloud forensics is of great significance for safeguarding the cloud computing security. The current cloud forensics technology research is at an early stage, and cloud forensics is faced with problems such as lack of digital evidence integrity, high forensics overhead and low intelligence. Therefore, an intelligent cloud forensics method based on SDS (software defined security) and cloud forensics trend analysis is proposed to mitigate some of these problems. Firstly, a cloud forensics architecture based on software defined security is proposed to realize collaborative real-time forensics between cloud network and cloud computing platform. Secondly, a cloud forensics trend analysis algorithm based on the HMM (hidden Markov model) is proposed to realize intelligent forensics strategy decision-making and forensics resource scheduling in the cloud forensics architecture. The experimental results show that, compared with the separate

收稿日期:2019-06-11;**修回日期**:2019-08-05

基金项目:江西省经济犯罪侦查与防控技术协同创新中心开放基金资助课题(JXJZXTCX-007,JXJZXTCX-009);国家重点研发计划项目(2016QY01W0200);广州市科技计划项目(201802020015);羊城创新创业领军人才支持计划资助(领军人才 2016008)

This work was supported by the Collaborative Innovation Center for Economics Crime Investigation and Prevention Technology, the Jiangxi Province (JXJZXTCX-007, JXJZXTCX-009), the National Key Research and Development Program of China (2016QY01W0200), the Science and Technology Planning Project of Guangzhou Municipality of China (201802020015), and the Support Scheme of Guangzhou for Leading Talents in Innovation and Entrepreneurship (2016008).

通信作者:丁丽萍(dingliping@gz.iscas.ac.cn)

network forensics method and cloud computing platform forensics method, the forensics capacity of this method increases to 91.6%, and the forensics overhead of this method is in between, achieving a better effect between forensics capability and forensics overhead. This method has some referential significance for cloud service providers to provide cloud forensics service.

Key words cloud computing; cloud forensics; digital forensics; software defined security (SDS); hidden Markov model (HMM); cloud forensics trend

摘 要 随着云计算的发展与普及,云计算环境下的安全问题日益突出.云取证技术作为事后追责与惩治技术手段,对维护云计算环境安全具有重大意义.云取证技术研究发展尚处于早期,云取证面临电子证据不完整、取证开销较大、取证过程智能化不足等难题.为缓解这些问题,提出一种基于软件定义安全(software defined security, SDS)和云取证趋势分析的智能云取证方法.首先,提出一种基于软件定义安全的云取证架构,实现云网络与云计算平台协同实时取证.其次,提出基于隐 Markov 模型的云取证趋势分析算法,实现云取证架构中的智能取证策略决策和智能取证资源调度.实验结果表明:相较于单独的网络取证与云计算平台取证,该方法取证能力提高至 91.6%,而取证开销则介于两者之间.该方法对云服务商提供云取证服务具有广泛的借鉴意义.

关键词 云计算;云取证;电子数据取证;软件定义安全;隐 Markov 模型;云取证趋势

中图法分类号 TP393.08

随着云计算的发展与普及,云计算环境下的安全问题日益突出.国家互联网应急中心发布的《2018我国互联网网络安全态势综述》[1]中提到云计算平台成为发生网络攻击的重灾区,云计算平台上的分布式拒绝服务攻击(distributed denial of service, DDoS)次数、被植入后门的网站数量、被篡改网站数量在各类型网络安全事件中占比均超过 50%.国内主流云计算平台上承载的恶意程序、木马和僵尸网络在境内互联网上占比超过 50%,可见目前国内云计算平台上的安全形势异常严峻,云安全问题成为阻碍云计算发展的关键因素.然而,无论事前防护做得多么完备,云安全问题都不可能杜绝,为了维护云服务提供商、云用户的正当权益,威胁打击云计算环境下各类违法犯罪活动,除了常见的安全防护手段,更需要云取证技术手段来进行事后追责和惩治.

电子数据取证是指科学地运用提取和证明方法,对于从电子数据源提取的电子证据进行保护、收集、验证、鉴定、分析、解释、存档和出示,以有助于进一步的犯罪事件重构或者帮助识别某些与计划操作无关的非授权性活动[2].云取证是对云计算环境中各类违反犯罪活动进行电子数据取证的过程,是电子数据取证技术在云计算环境这样一个特定的场景下的应用[3],云取证可以细分为网络取证、云计算平台取证和云终端取证.网络取证是抓取、记录和分析网络事件以发现安全攻击或安全事件的来源,网络取证的目的是保护用户和资源、防范因网络连接和数据传输而产生的被非法利用、入侵以及其他犯罪行为[4].云计算平台取证按照云服务模式可分为基础设施即服务(infrastructure as a service, IaaS)取证、平台即服务(platform as a service, PaaS)取证和软件即服务(software as a service, SaaS)取证,主要是对被攻击的云用户系统所在虚拟机进行取证,目的是维护云服务提供商、云用户的正当权益,威胁打击云计算环境下各类违法犯罪活动.云终端取证是对云用户客户端的取证,客户端存有一些缓存数据和离线数据,这些数据也有助于案情分析,一般作为云取证的一种补充手段,目的是尽可能多地提取涉案电子证据.本文从云服务商的角度出发,在下文中提及的云取证主要涉及云网络取证和云计算平台取证.总之,云取证需要云网络取证与云计算平台取证相结合才能最大限度地保证电子证据的完整性.

云取证技术发展尚不完善,云取证技术面临诸多难题:

1) 云计算环境中 80%的网络流量是云计算环境内部的东西向流量,部署在外部交换机上的网络取证设备无法捕获东西向流量,导致网络取证设备提取的电子数据不完整.

2) 云计算环境下数据采用分布式存储,数据可能分散存储于不同的物理数据中心,而这些数据中

心可能位于不同的司法管辖范围内,这种情况下针对物理设备的事后取证方法面临数据定位和提取困难[5].

3) 虚拟化技术会频繁的回收和再分配各类资源,导致云计算平台中的数据成为易失性数据,一旦被释放回收,将难以恢复,导致传统事后取证方法提取的电子数据不完整[5].

4) 由于云计算中资源的所有权、管理权和使用权的分离使得云用户失去了对物理资源的直接控制,这其中也包括对网络环境的控制与访问,导致云用户无法在网络层进行取证[5].

5) 云计算环境的证据提取技术智能化程度较低,大多靠手动取证,人力成本大、效率低下,使得云取证技术在实际应用中更加困难.

可以看出,云计算环境为取证带来的难题主要来自数据获取和取证效率等方面.本文针对这些难题,提出一种基于软件定义安全和云取证趋势分析的智能云取证方法,主要贡献有 4 个方面:

1) 提出基于软件定义安全的云取证架构(software defined security based cloud forensics framework, SDS-CF),借鉴软件定义安全的分层理念,将取证分为数据层、控制层与应用层.一方面通过分层实现入侵检测和云取证等安全服务的集中管理,在此基础上实现云网络与云计算平台的协同实时取证,在安全事件发生的第一时间进行在线电子证据提取、保全与存证,从而有效规避事后取证方法面临的种种难题.一方面通过对软件定义网络(software defined network, SDN)网络控制器的引流控制,实现云网络东西向流量的取证.一方面通过分层实现可定制化云网络取证服务,满足云用户网络取证需求.

2) 提出了云取证趋势的概念并进行形式化定义,云取证趋势是指在云网络环境中获取并理解云网络安全事件告警信息,从而实时预测云取证趋势,为实现智能云取证提供了理论支持.

3) 提出了基于隐 Markov 模型(hidden Markov model, HMM)的云取证趋势分析算法和基于改进告警质量的入侵检测系统(intrusion detection system, IDS)告警选择算法,将该算法运用于 SDS-CF 中进行取证策略智能决策和取证资源智能调度,实现智能云取证.同时,为了对取证趋势分析算法的效果进行量化评估,提出了取证能力和取证开销 2 个指标.

4) 在仿真云计算平台上基于林肯实验室的经典数据集 LLDOS1.0 进行取证实验,通过实验结果证明基于软件定义安全和云取证趋势分析的智能云取证方法取证能力相比单独的网络取证或者云计算平台取证提高至 91.6%,取证开销则介于两者之间,在取证能力与取证开销之间取得较好平衡.该方法对云服务商提供云取证服务具有广泛的借鉴意义.

1 相关工作

本文从云服务商的角度出发,云取证主要涉及网络取证和云计算平台取证.本节将对这 2 方面的工作进行概述.

网络取证技术出现在云计算以前,研究相对比较成熟,提出的研究方法也很多,文献[6-7]中均有较为详细的综述.其中与本文相关的研究工作包括基于入侵检测的网络取证技术研究和网络攻击溯源取证技术研究.

在进行网络取证时,何时触发取证系统进行数据的提取非常关键,决定网络取证是否满足电子证据的完整性.为此,网络取证常与网络监控相结合,其中又以入侵检测告警为主要的取证判断依据[8].此外,入侵检测告警本身也是网络取证分析的对象;文献[9]提出基于入侵检测的网络取证方法,该方法基于静态分析和动态分析相结合的多维取证分析,对提取到的网络日志进行分析,从而识别出数据包中的恶意行为;文献[10]提出基于统一威胁管理和入侵检测探针的协同网络安全管理系统,将多个子网探针的入侵检测告警数据汇集起来进行取证分析,能有效地检出僵尸网络和 DDoS 攻击并进行取证分析;文献[11]提出了一个入侵检测分析架构和一个概率推理机制,通过这个推理机制对入侵检测告警进行解释,并自动生成最大似然取证分析报告.

识别攻击数据包源头的过程叫做溯源,又叫 IP 地址溯源,溯源取证对 DDoS 攻击、IP spoofing 攻击和僵尸网络的取证分析非常有效[12-14];早期溯源取证多基于网络数据包标记[15-17]或者网络设备日志分析[18],溯源分析效率较低;而随着 SDN 网络的流行[19],为溯源取证提供了极大的便利,因为 SDN 网络控制器可以作为一个观测点给出网络流的全局图[20];因此很多学者基于 SDN 网络提出了新的网络取证分析方法,文献[21]提出了一个基于 SDN 网络的有向图模型,通过该图模型能检测一条异常流在该 SDN 网络中的入口点,也就是第一步路由,且不用监控 IP 地址就能找到和攻击相关的所有的流,

该模型提高了 SDN 网络的溯源取证效率;文献[22]提出在 SDN 网络增加一个取证管理层(forensics manage layer, FML),通过 FML 对 SDN 网络中的各种攻击进行实时分析和取证,从而降低 SDN 控制器的分析负载;文献[23]则提出一种 SDN 网络环境下的支持反取证的攻击模型,该攻击模型基于 SDNMap 扫描器实现,能够在不知道 SDN 控制器信息和网络结构的前提下还原流规则和网络策略,并通过这些信息构造巧妙的攻击流绕过访问控制列表(access control list, ACL)机制和取证设备.

云计算平台取证技术是随着云计算的发展而发展[24-25].其中与本文相关的研究包括云取证框架的研究和云计算平台日志取证研究.

Ruan 等人[26]在全球范围发起了一项关于云取证关键问题的调查,基于 257 位电子数据取证专家的问卷调查,超过八成的专家认为云取证领域最主要的研究方向是设计云取证模型.文献[27-28]围绕基于不同云服务模式的云取证模型研究进行了综述.文献[29]提出一种 IaaS 云服务模型下的取证框架 ICFF (IaaS cloud forensics framework),通过在虚拟机中安装轻量取证代理进行电子证据的提取,并设计取证虚拟机用于证据存储、保全和分析,保障证据数据的完整性及机密性;文献[30]提出在 IaaS 模式下通过虚拟机自省进行可疑行为发现和取证分析,具体使用到 Swap 分区分析,连续交付模式下的数据窥探、终止进程分析等技术;文献[31]提出一种基于隐藏事件触发机制的内存取证方法 ForenHD,该方法也是利用虚拟机自省技术来监视虚拟机中的内核对象,当发现了隐藏对象时则出发取证机制,提取代码段等信息.

基于云计算平台日志分析的方法依然是目前最主流的云计算平台取证分析方法.文献[32]分析了目前主流的云计算平台日志分析方法,得出主要难点是云日志的获取、云日志的选择、云日志数据的完整性和可信性问题,而且云日志取证分析依赖于云服务提供商;文献[33]提出一种 Secure-Logging-as-a-Service 的云取证服务,通过实时存储虚拟机日志以保证日志的可靠性,并作为一种取证服务向司法鉴定人员开放;文献[32]为了保护云用户日志的可靠性和隐私性,提供一种取证模型 CLASS(cloud log assuring soundness and secrecy scheme),在实时提取云用户日志的同时通过云用户公钥来加密云用户日志,使得云用户的隐私得到保护,且采用 PPL(proof of past log)算法来保护日志的完整性.

虽然网络取证和云计算平台取证的相关研究工作比较丰富,但是网络取证方面针对云网络环境的取证研究较少,缺乏东西流量的网络取证方法,云计算平台取证方面,针对智能取证的研究较少.且在实际工作中我们发现将网络取证与云计算平台取证分离开来,一方面会损失掉一些潜在的证据信息,不利于整个网络犯罪过程的重建;另一方面取证开销较大,不利于实时取证.

本文提出一种基于软件定义安全和云取证趋势分析的智能云取证方法,一方面,通过借鉴软件定义安全的分层理念实现云网络与云计算平台协同实时取证;另一方面,基于隐 Markov 模型的云取证趋势分析算法实现智能云取证,该方法在取证能力与取证开销之间取得较好平衡,该方法为解决目前云取证面临的电子证据不完整、取证开销较大、取证过程智能化不足等挑战提供了一种思路.

2 基于软件定义安全的云取证架构

本节提出了基于软件定义安全的云取证架构(software defined security based cloud forensics framework, SDS-CF),介绍了软件定义安全理念,分析了 SDS-CF 的 3 层架构,总结了 SDS-CF 主要特点,并进一步分析了取证管理模块工作流程.

2.1 软件定义安全

软件定义安全(software defined security, SDS)[34]这一理念从软件定义网络[35]引申而来,其架构如图 1 所示,通过安全数据层与控制层分离,将各类网络安全设备的接入模式、部署方式、实现功能进行解耦.在数据层为各类安全设备定义标准的接口,屏蔽各个厂商同类安全设备的差异性,使得在控制层能够对安全设备进行统一的管理和编程.应用

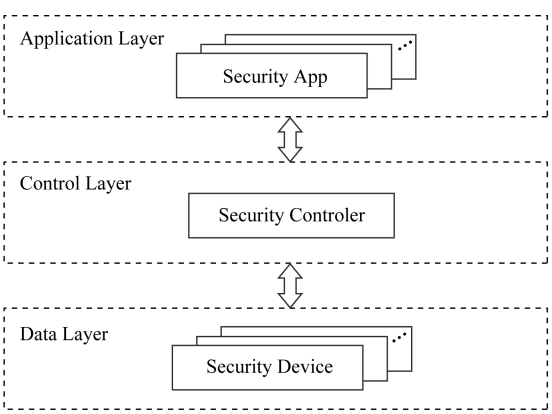


Fig. 1 The SDS framework

图 1 SDS 架构图

层支持对安全设备业务进行软件编程,从而实现智能化、自动化的安全业务编排和管理,以定制较为复杂的自动化的安全服务链.

文献[34,36]总结了软件定义安全架构的3个核心特征:

1) 自动化和智能化.软件定义安全架构将控制和管理进行抽象并集中在控制层上,从而实现安全设备业务的可编程.

2) 开放API.安全设备提供开放的API接口打破了不同厂商安全设备互不兼容的局限,能有效增强不同安全设备厂商、不同安全设备之间的协作性.

3) 安全服务的编排.通过把原本相互独立的、单一的安全服务进行有效的编程组合,编排成一系列安全服务链,实现自动安全运维.

软件定义安全契合了云取证的部分需求:控制层与数据层的分离,一方面使得数据能快速聚合提高云取证的自动化程度;另一方面取证本身也可以作为安全服务的一种通过控制层的统一封装为用户提供定制取证服务;再一方面可以参与到安全服务的编排中,实现检测、应急响应与取证的安全全生命周期服务链.

但软件定义安全架构有存在一些局限:一方面该架构接入防护对象网络,只能提供网络流量的取证服务,无法提供云计算平台的取证能力;另一方面该机制无法满足电子证据可采性规则.

2.2 SDS-CF的3层架构

软件定义安全理念一定程度上契合了云取证的需求,但无法完全满足云取证的需求.为此我们借鉴软件定义安全理念,结合网络取证技术、虚拟机取证技术、区块链存证技术等提出云取证架构SDS-CF.

SDS-CF遵循软件定义安全的层次结构,从下往上分为数据层、控制层和应用层,如图2中虚线框内所示:

1) 数据层也称为基础设施层,在这一层,各种

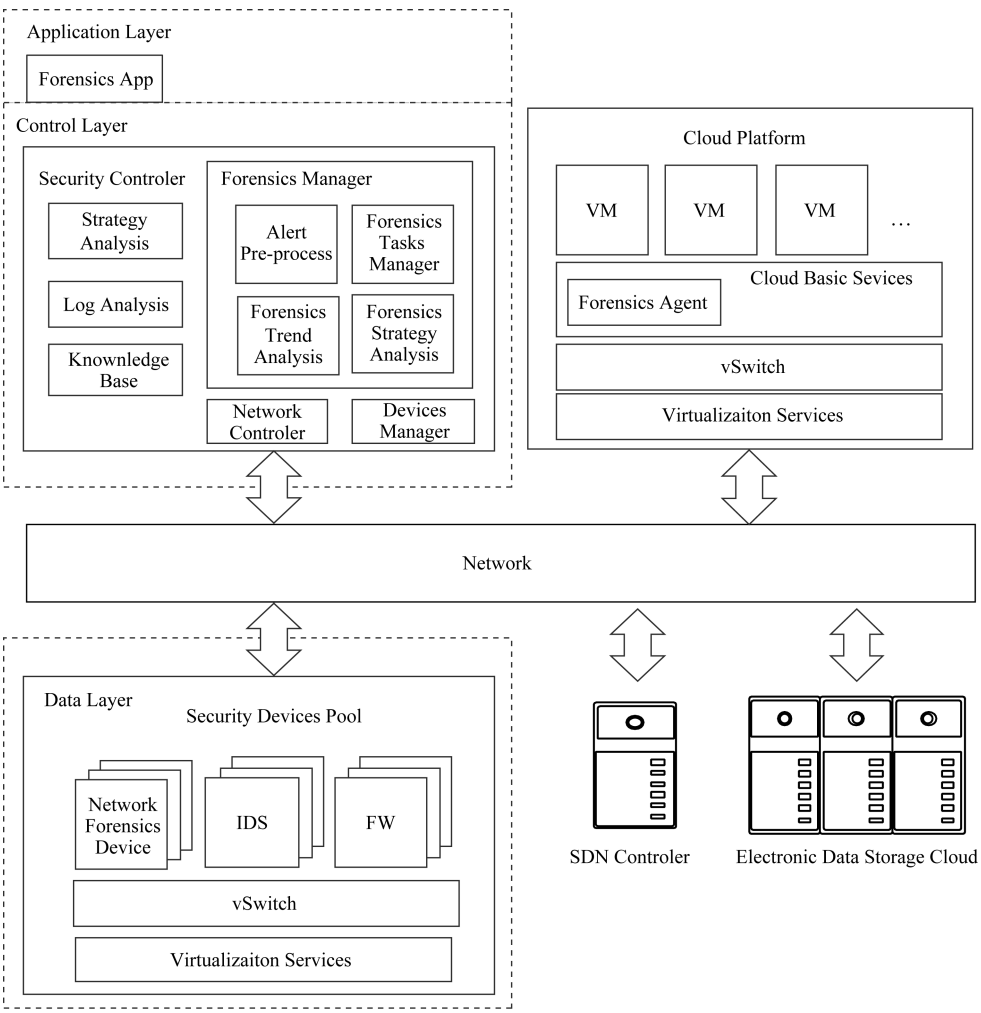


Fig. 2 The hierarchy graph of SDS-CF

图2 SDS-CF层次图

安全设备被抽象出来,并通过接口供控制层面灵活调度,网络取证设备和其他安全设备被池化以提供基本的安全服务,网络取证设备可根据具体业务动态调整服务能力,以满足不同取证场景的需求.数据层和控制层通过接口交互取证任务和网络数据流等信息.

2) 控制层是该架构的核心,主要负责取证管理、安全设备管理、各类安全策略决策和执行.一方面与云计算平台对接,通过取证管理模块控制云计算平台中的取证服务代理[29]协同执行取证任务;另一方面负责与 SDN 网络控制器对接,获取全局流信息用于攻击溯源,并能控制网络流量,将云计算环境中南北向流量和东西向流量镜像至网络取证设备.一方面通过网络与区块链存证平台连接,将实时提取的电子数据的五要素(取证的时间、取证的地点、电子数据的 Hash 值、取证设备编号、取证人员)存至存证平台,利用区块链技术[37]保障云取证的真实性.

3) 应用层将取证服务抽象成服务单元,可根据云用户需求提供定制的云网络取证服务.应用层还可将取证服务加入安全服务链编排[38],实现云计算环境下检测、应急响应与取证的安全全生命周期服务链.

2.3 SDS-CF 的特征

通过 SDS-CF 的 3 层架构概述总结出该架构有 3 个特征:

1) 支持云网络与云计算平台实时协同取证.一方面通过分层实现入侵检测和云取证等安全服务的集中管理,在此基础上实现云网络与云计算平台的协同实时取证,在安全事件发生的第一时间进行在线电子证据提取、保全与存证,从而有效规避事后取证方法面临的种种难题.

2) 支持云计算环境内东西向流量取证.通过对 SDN 网络控制器的引流控制,实现云网络东西向流量的取证.

3) 满足电子证据可采性规则,要求取证满足真实性、相关性与和合法性.该架构通过云网络与云计算平台的实时协同取证能提取完整的相关电子证据,满足取证相关性;该架构基于区块链存证平台实现电子证据的实数存证,满足云取证的真实性,并为日后出具的司法鉴定报告的合法性提供支撑.

2.4 云取证管理模块工作流程

SDS-CF 最核心的是取证管理模块.其取证管理工作流程大致如图 3 所示:

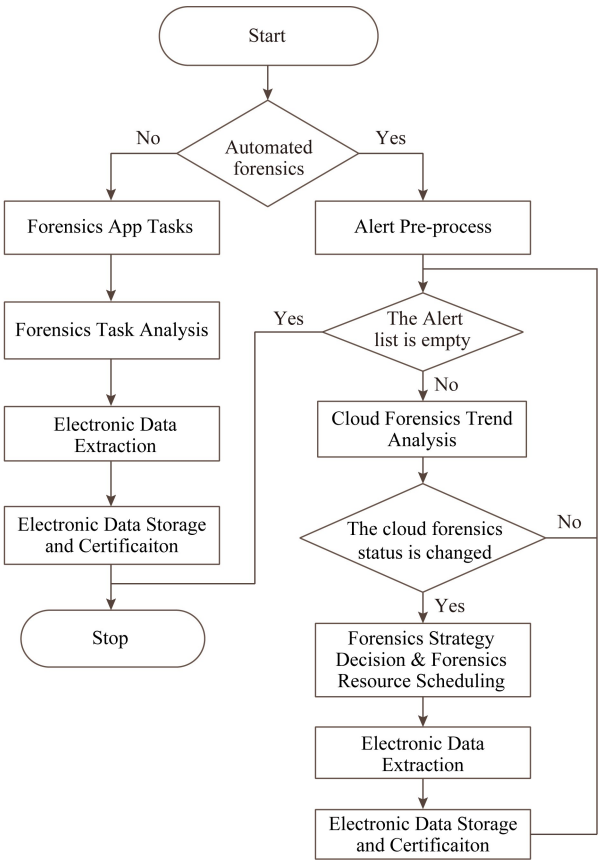


Fig. 3 The workflow of cloud forensics of SDS-CF

图 3 SDS-CF 的云取证工作流程图

取证管理流程大致包括 3 个部分:

1) 云取证服务的触发

云取证服务可以由云用户发起,也可以由入侵检测告警自动触发.由云用户主动发起的取证服务往往已经设置好取证各项参数,通过解析之后进入取证环节.而入侵检测告警触发的自动取证流程则要复杂得多,因 IDS 具有告警频繁、数量庞大和误报率高等特征,需要通过下文将提出的云取证趋势分析阶段来分析云取证状态和预测云取证趋势,根据不都能通过的取证状态出发取证服务.

2) 云取证策略制定与取证资源调度

根据云取证趋势分析算法进行取证策略智能决和取证资源智能调度,以提高取证准确度和降低取证开销.

3) 电子数据的提取与保全

网络取证设备和云计算平台取证代理根据取证策略进行电子数据提提取,并实时生成电子证据五要素(取证的时间、取证的地点、电子证据的 Hash 值、取证设备编号、取证人员)提交至区块链存证云进行保全.

3 基于 HMM 的云取证趋势分析算法与应用

为了实现智能云取证,提高云取证效率.本节提出云取证趋势概念,基于隐 Markov 模型进行云取证趋势形式化定义,并提出云取证趋势分析算法.为了对取证趋势分析算法中的观测序列选择进行优化,提出基于改进告警质量的 IDS 告警选择算法.通过将云取证趋势分析算法应用于第 2 节中提出的 SDS-CF 架构实现智能取证.同时,为了对该算法的效果进行评估,本节提出取证能力和取证开销 2 个量化评估指标.

3.1 基于 HMM 的云取证趋势形式化定义

云取证趋势是指在云网络环境中获取并理解云网络安全事件告警信息,从而实时预测云取证趋势.云取证趋势被 SDS-CF 用于取证策略智能决策和取证资源智能调度.

隐 Markov 模型[39]是一种统计分析模型,能用来描述一个双重随机过程,它具有一定状态数的隐 Markov 链和显示随机函数集.它的状态不能直接观察到,但能通过观测序列观察到,每个观测序列都是通过某些概率密度分布表现为各种状态,每一个观测序列是由一个具有相应概率密度分布的状态序列产生[40].云取证趋势是由云网络攻击驱动的取证需求的累积,取证趋势的计算来源于对网络攻击的行为的观测,因此本文提出基于 HMM 的云取证趋势分析模型.相关定义为:

定义 1. 云取证状态.云取证状态由集合 S 表示:

$$S=\{s_1,s_2,s_3,s_4\}, \tag{1}$$

其中,s_1 表示不取证状态;s_2 表示网络取证状态,此状态下将对网络设备日志进行提取;s_3 表示协同实时取证状态,此状态下将对网络设备日志、网络流量镜像、云计算平台日志、虚拟机日志进行提取;s_4 表示全量取证状态,此状态下将对网络设备日志、网络流量镜像、云计算平台日志、虚拟机镜像进行提取.

定义 2. 云网络安全事件.云网络安全事件由集合 V 表示:

$$V=\{v_1,v_2,v_3,v_4\}, \tag{2}$$

其中,v_1 表示无安全事件;v_2 表示扫描类安全事件,包括 DDoS 攻击;v_3 表示入侵类安全事件;v_4 表示提权类安全事件.

定义 3. 云取证状态概率模型.云取证状态概率由隐 Markov 模型表示:

$$\lambda=(S,V,\boldsymbol{P},\boldsymbol{Q},\boldsymbol{\pi}), \tag{3}$$

其中,$\lambda_t(i)$ 表示在时刻 t 时云取证状态为 s_i 的概率;S 为云取证状态集合空间,取值范围见定义 1;V 为观测序列集合空间,表示观测到的安全事件的样本空间,取值范围见定义 2;$\boldsymbol{P}$ 为不同云取证状态转移概率矩阵,$\boldsymbol{P}=(p_{ij})$,

$$p_{ij}=prop(q_{t+1}=S_j\,|\,q_t=S_i),1\leqslant i,j\leqslant N, \tag{4}$$

其中,p_{ij} 表示在时刻 t 时云取证状态处于 s_i 状态且在时刻 $t+1$ 云取证状态处于 s_j 的概率;$\boldsymbol{Q}$ 为观测序列的概率分布矩阵,$\boldsymbol{Q}=(q_{ij})$,

$$q_{ij}=prop(o=v_j\,|\,q_t=s_i),1\leqslant i\leqslant N,1\leqslant j\leqslant M, \tag{5}$$

其中,q_{ij} 表示时刻 t 云取证状态为 s_i 且观测到的安全事件为 v_j 的概率;$\boldsymbol{\pi}$ 为初始状态概率分布矩阵,$\boldsymbol{\pi}=(\pi_i)$,

$$\pi_i=prop(q_1=s_i),1\leqslant i\leqslant N, \tag{6}$$

其中,π_i 表示在时刻 $t=1$ 时云取证状态为 s_i 的概率.

定义 4. 观测序列.观测序列由集合 O 表示,指网络安全设备在评估时间段 T 内输出的告警信息按定义 2 进行归类后构成的告警序列,表示形式为

$$O=\{o_1,o_2,\cdots,o_n\}, \tag{7}$$

其中,o_i 表示在时刻 t_i 观测到的安全事件,$o_i\in V$,$T=\{t_1,t_2,\cdots,t_n\}$.

定义 5. 云取证趋势.云取证趋势由 γ_t 表示,指在时刻 t 时观测到告警序列 $O=\{o_1,o_2,\cdots,o_t\}$,预测当前的云取证趋势:

$$\gamma_t=\sum_{i=1}^{N}\lambda_t(i)C(i), \tag{8}$$

其中,$1\leqslant i\leqslant N$,$C(i)$ 表示云取证状态 s_i 下的取证代价系数.本文中 $C(i)$ 以取证存储开销为依据制定.因不同取证状态的取证数据范围不同,对应的取证计算开销和存储开销也不同.在云取证实务中存储开销是最为关键的指标之一,取证代价系数 $C(i)$ 以取证存储开销为依据,通过大量真实取证案例统计得出,将在实验环节进一步分析.

3.2 云取证趋势分析算法

为了计算在时刻 t 的云网络取证趋势 γ_t,主要采用经典的前向算法[40].

1) 确定观测序列 O 和模型参数 $\lambda=(S,V,\boldsymbol{P},\boldsymbol{Q},\boldsymbol{\pi})$.

2) 计算在时刻 t 且云取证状态为 s_i 时的概率 $\lambda_t(i)$,计算过程示意图如图 4 所示.

在时刻 t 时各云取证状态的概率 $\lambda_t(i)$ 计算为

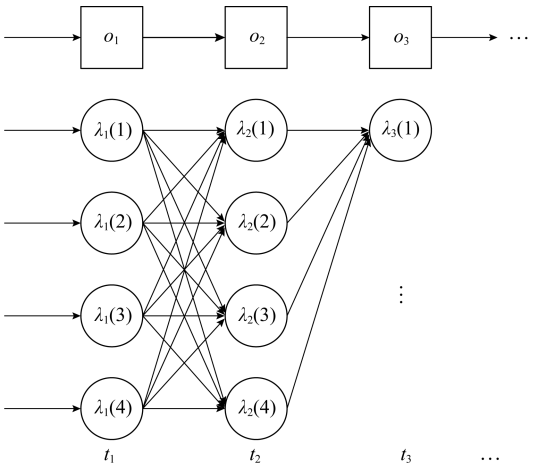


Fig. 4 The schematic diagram of forward algorithm

图 4 前向算法示意图

$$\lambda_t(i)=(q_i(o_t)\sum_{j=1}^{N}\alpha_{t-1}(j)p_{ji})\Big/\sum_{i=1}^{N}(q_i(o_t)\sum_{j=1}^{N}\alpha_{t-1}(j)p_{ji}),\tag{9}$$

其中,$1\leqslant i,j\leqslant N$,$N$ 是状态数目;当 $t=1$ 时,$q_i(o_1)=\pi_i$.

3) 计算在时刻 t 的云取证趋势 γ_t,计算方法见式(8).

3.3 基于改进告警质量的告警选择算法

云取证状态概率模型中的观测序列采用 IDS 告警,通过云取证趋势分析算法可知,观测序列的质量对云取证趋势分析算法的准确度至关重要[41].而入侵检测系统往往会产生海量的告警,且误报率较高.为了能从海量的告警中提取出高质量的报警信息,本文从取证的角度出发提出改进的告警质量算法(revised quality of alert, rQoA)进行告警的筛选.告警质量算法(quality of alert, QoA)[42]是从安全态势分析的角度出发,对 Snort 原始告警信息分段处理,在每个片段内依据告警出现的频率(alert frequency, AlF)、告警关键程度(alert criticality, AlC)和告警严重程度(alert severity, AlS)对告警信息进行综合评估,并从中选取质量最高的告警作为该片段的观测序列.然而在取证实务中更注重事件的关联性,QoA 算法可能导致一些告警质量不高的关联事件告警的遗漏从而影响取证趋势的判断.为此本文提出的 rQoA 算法引入告警关联性影响因子(alert relevancy, AlR)来替换 AlC 这一维度,能有效地将关联事件筛选出来.

为了计算 AlR,从源 IP 地址相关性、目的 IP 地址相关性、告警 IP 链相关性、反告警 IP 链相关性、源端口相关性、目的端口相关性、时间相关性、告警类型相关性 8 个维度进行计算,并对每个相关性赋予不同的权重,取值参考文献[43].

在时刻 t 时,观测序列 o_t 的告警关联性影响因子 $\varphi(o_t)$:

$$\varphi(o_t)=\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{M}\sigma_j f_j(o_t,o_{t-i})\Big/\sum_{j=1}^{M}\sigma_j\right),\tag{10}$$

其中,$f_j(o_t,o_{t-i})$表示事件 o_t 和 o_{t-i}的属性 j 的关联度,σ_j 表示属性 j 关联度的权重,M 表示关联属性的个数,从上文可知 $M=8$,N 表示往回对比的观测序列的维度,此处 $N=3$,也就是对比最近连续 3 件告警事件的关联度.

为了统一到值域[1,3],进行离差标准化的反函数进行处理:

$$AlR=\varphi(o_t)(max-min)+min.\tag{11}$$

最后计算告警质量:

$$rQoA=\frac{1}{2}(AlF\times AlS+AlF\times AlR+AlR\times AlS),\tag{12}$$

3.4 云取证趋势分析算法在 SDS-CF 中的应用

云取证趋势分析算法作为 SDS-CF 进行取证管理模块的核心算法,主要用于取证策略智能决策与取证资源智能调度.通过对云取证状态的持续分析,SDS-CF 能实现动态地调整取证策略和调度取证资源,从而实现智能的云取证服务.

其工作原理与评估指标为:

1) 取证策略智能决策

SDS-CF 控制器中的取证管理模块根据 IDS 告警信息持续分析云计算环境所处的取证状态,根据当前所处的云取证状态 S 进行取证策略决策,并根据状态变化动态调整取证策略.

定义 6. 取证策略.取证策略由矩阵 $\boldsymbol{D}$ 表示:

$$\boldsymbol{D}=\{d_1,d_2,d_3,d_4\},\tag{13}$$

其中,d_i 表示当云取证状态为 s_i 时的取证策略,由五元组表示,表示形式:

$$d_i=(f_1,f_2,f_3,f_4,f_5),\tag{14}$$

其中,f_1 表示网络日志;f_2 表示网络流量;f_3 表示云计算平台日志;f_4 表示虚拟机日志;f_5 表示虚拟机镜像.其取证范围为{0,1},当 $f_i=0$ 时表示不对该数据源进行取证,当 $f_i=1$ 时表示需要对该数据源进行取证.

依据定义 1,提出云取证策略 $\boldsymbol{D}$ 取值:

$$\boldsymbol{D}=\begin{pmatrix}0&0&0&0&0\\1&0&0&0&0\\1&1&1&1&0\\1&1&1&1&1\end{pmatrix},\tag{15}$$

对应的云取证策略智能决策过程为:

在时刻 t,根据云取证状态概率模型计算得出:

当云取证状态最大概率处于 s_1 时,采取取证策略为 d_1;

当云取证状态最大概率处于 s_2 时,采取取证策略为 d_2;

当云取证状态最大概率处于 s_3 时,采取取证策略为 d_3;

当云取证状态最大概率处于 s_4 时,采取取证策略为 d_4.

2) 取证资源智能调度

SDS-CF 控制器中的取证管理模块通过云取证趋势分析实时预测取证开销,从而依据取证开销调度合适的取证资源.依据 3.2 节算法描述可知,可根据不同的资源开销来计算云取证趋势,而本文主要从存储资源开销的角度进行分析,因此 SDS-CF 的设备管理模块也以存储能力对安全资源池中的网络取证设备进行管理,并通过取证趋势分析结果在网络取证设备资源池中调度满足取证存储需求的网络取证设备实现取证资源的智能调度.

3.5 取证能力指标与取证开销指标

为了对云取证趋势分析算法的效果进行评估,针对取证策略决策的效果和取证资源调度的效果分别提出取证能力和取证开销评估指标.

定义 7. 取证能力.取证能力(capacity of forensics, CoF)指在观测时间段 T 内采用的取证策略能覆盖所有安全事件相关的电子数据的概率.

根据定义 6 可得:不同云取证状态下的取证策略的包含关系为 $d_1\subset d_2\subset d_3\subset d_4$,则随着云取证状态的递增,其取证能力是累加的.

$$CoF=\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{i_{\max}}\lambda_t(i),\tag{16}$$

其中,$1\leqslant i\leqslant i_{\max}$,$1\leqslant t\leqslant T$,$T$ 表示观测时长,$\lambda_t(i)$ 见定义 3,$i_{\max}$ 表示在时刻 t 时概率最大的云取证状态为 $s_{i_{\max}}$.

定义 8. 取证开销.取证开销(overhead of forensics, OoF)指在观测时间段 T 内的取证存储资源开销,且随着取证代价系数的变化而变化.计算为

$$OoF=\sum_{t=1}^{T}\gamma_t,\tag{17}$$

其中,$1\leqslant t\leqslant T$,T 表示观测时长,γ_t 取值见定义 5.

4 实验与结果

4.1 实验设计与实验环境

为了对本文提出的基于软件定义安全和云取证趋势分析的智能云取证方法进行验证,涉及 2 项实验:第 1 项实验为 SDS-CF 云取证流程验证实验;第 2 项实验为云取证趋势分析算法效果对比实验.实现在仿真云计算平台上基于林肯实验室的经典数据集 LLDOS1.0 的进行云取证实验.LLDOS1.0 是一个 DDoS 攻击场景的测试数据集,以离线的网络流量包形式提供.基于 LLDOS1.0 的网络流浪包构建一个仿真的云计算环境测试环境,在云计算平台中启动 3 台虚拟机并配置成受害者 IP,并将流量包在仿真环境中重放以模拟对仿真云计算平台的攻击.

4.2 云取证状态概率模型构建

1) 观测序列 $\boldsymbol{O}$ 的选取

基于 3.3 节提出的基于改进告警质量的 IDS 告警选择算法进行观测序列 $\boldsymbol{O}$ 的选取.通过对 LLDOS1.0 数据集进行分析,样本数据集时间约为 22:21:00—01:36:00 之间,设定划片时长 $\Delta t=5$min,整个数据集可大致划分为 40 个片段,对应的观测序列为 40 维.

2) 云取证状态概率模型 λ 构建

云取证状态概率模型参数确定需要资深安全专家经验,且因数据集采样时期的网络空间安全攻防形势和现在有所不同,状态转移矩阵 $\boldsymbol{P}$ 和为观测序列的概率分布矩阵 $\boldsymbol{Q}$ 均借鉴文献[42]使用的模型参数.且将云取证状态和网络安全状态[42]作一一对应,从而得到参数:

$$\boldsymbol{P}=\begin{pmatrix}p_{11}&p_{12}&p_{13}&p_{14}\\p_{21}&p_{22}&p_{23}&p_{24}\\p_{31}&p_{32}&p_{33}&p_{34}\\p_{41}&p_{42}&p_{43}&p_{44}\end{pmatrix}=\begin{pmatrix}0.839&0.15&0.009&0.002\\0.005&0.972&0.02&0.003\\0.004&0.017&0.975&0.004\\0.004&0.017&0.125&0.854\end{pmatrix},\tag{18}$$

$$\boldsymbol{Q}=\begin{pmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \end{pmatrix}= \begin{pmatrix} 0.8999 & 0.02 & 0.08 & 0.0001 \\ 0.6699 & 0.25 & 0.08 & 0.0001 \\ 0.735 & 0.1 & 0.16 & 0.005 \\ 0.8 & 0.04 & 0.11 & 0.05 \end{pmatrix}. \tag{19}$$

初始状态概率分布矩阵 $\boldsymbol{\pi}$:

$$\boldsymbol{\pi}=(\pi_1\ \ \pi_2\ \ \pi_3\ \ \pi_4)=(1\ \ 0\ \ 0\ \ 0). \tag{20}$$

4.3 实验 1. SDS-CF 云取证流程验证

本实验将针对 LLDOS1.0 的攻击过程给出对应的 SDS-CF 云取证过程,进行 SDS-CF 工作原理的验证.

通过对数据集的分析可知[44],LLDOS1.0 攻击过程为:

1) 阶段 1.22:51:36—22:52:00,攻击者(202.77.162.213)对目标网络进行扫描以找寻活跃主机.

2) 阶段 2.23:08:07—23:18:05,攻击者利用 ping 协议探测活跃主机是否存在 sadmind 服务,并选取了 Mill(172.16.115.20)作为傀儡机.

3) 阶段 3.23:33:10—23:35:01,攻击者利用 sadmind 服务的缓存区溢出漏洞渗透进目标系统.

4) 阶段 4.23:50:01—23:50:54,攻击者在受害者机器上安装 DDoS 攻击代码和木马后门.

5) 阶段 5.00:26:15—00:34:21,攻击者通过后门控制 Mill 对攻击目标(131.84.1.31)发起 DDoS 攻击.

其中,SDS-CF 使用 Snort 作为 IDS,Snort 版本为 2.9.9.0,检测出各阶段的报警数量如表 1 所示:

Table 1 Statistic of LLDOS1.0

表 1 LLDOS1.0 数量分析

Attack Stage	Packages Statistics	Snort Alert Statistics
Stage1	1 293	22
Stage2	34 034	100
Stage3	4 497	38
Stage4	2 249	9
Stage5	96 313	123

SDS-CF 的对应的云取证过程如下:

SDS-CF 以 $\Delta t=5$min 为时间窗口对取证策略和取证资源调度进行动态调整,是一个连续的过程.通过 2.3 节可知,在每一个 Δt 中,SDS-CF 中的取证管理模块工作过程为:

1) 通过告警预处理分析 IDS 告警序列并调整当前云取证状态,根据云取证状态触发或者更新取证流程.

2) 通过取证趋势分析,得出当前大概率所处的取证状态和取证趋势,生成对应的取证策略,并向设备管理模块发出指令,从取证资源列表中选择满足取证开销的网络取证设备并下发取证策略,同时根据想云计算平台取证代理模块发送取证指令和取证策略.

3) 向网络控制模块发出指令,将可疑流量镜像至网络取证取证设备.

4)取证管理模块读取网络取证设备和云计算平台取证代理提取的电子数据的五要素发送至存证云进行存证.

针对 LLDOS1.0 攻击过程的 5 个阶段,SDS-CF 对取证策略和取证资源做调整:

1) 阶段 1.通过云取证趋势分析,当前取证状态大概率处于 s_2,根据 3.4 节采取取证策略 $d_2=(1\ 0\ 0\ 0\ 0)$,通过表 2 可知当前取证趋势约为 0.6,向设备管理模块发出指令,从取证资源列表中选择满足取证开销的网络取证设备,并下发取证策略.

2) 阶段 2.通过云取证趋势分析,当前取证状态大概率处于 s_2,取证状态没变化,取证策略也不发生变化,通过表 2 可知当前取证趋势约为 0.8,向设备管理模块发出指令,检查当前网络取证设备满足取证要求,不更新网络取证设备.

3) 阶段 3.通过取证趋势分析,当前取证状态大概率处于 s_3,根据 3.4 节采取取证策略 $d_3=(1\ 1\ 1\ 1\ 0)$,通过表 2 可知当前取证趋势约为 1.5,向设备管理模块发出指令,检查当前网络取证设备满足取证要求,下发新的取证策略,同时向云计算平台取证代理发送取证策略与取值指令.

4) 阶段 4.通过取证趋势分析,当前取证状态大概率处于 s_4,根据 3.4 节采取取证策略 $d_4=(1\ 1\ 1\ 1\ 1)$,检通过表 2 可知当前取证趋势约为 66,向设备管理模块发出指令,当前网络取证设备不满足取证要求,从取证资源列表中选择满足取证开销的网络取证设备并下发取证策略,同时向云计算平台取证代理发送取证策略与取值指令.

5) 阶段 5.通过取证趋势分析,当前取证状态大概率处于 s_3,采取取证策略 $d_3=(1\ 1\ 1\ 1\ 0)$,通过表 2 可知当前取证趋势约为 2,根据 3.4 节向设备管理模块发出指令,检查当前网络取证设备满足取证要求,下发新的取证策略,同时向云计算平台取证代理发送取证策略与取值指令.

Table 2 The Results of Cloud Forensics Trend of CFS, NFS and AFS Strategies During the Attack Process of LLDOS1.0

表 2 LLDOS1.0 攻击过程的 CFS,NFS 和 AFS 三种策略云取证趋势结果

Timeline	CFS	NFS	AFS
22:25:00	0	0	0
22:30:00	0.291 656	0.022 035	2.056 450
22:35:00	0.255 145	0.043 542	3.563 366
22:40:00	0.123 624	0.034 636	2.511 979
22:45:00	0.588 322	0.096 251	8.737 096
22:50:00	0.179 844	0.053 934	4.448 913
22:55:00	0.104 596	0.036 124	2.649 047
23:00:00	0.088 196	0.028 891	1.918 483
23:05:00	0.083 616	0.025 977	1.624 094
23:10:00	0.082 063	0.024 805	1.505 790
23:15:00	0.081 481	0.024 335	1.458 306
23:20:00	0.081 253	0.024 147	1.439 257
23:25:00	0.539 129	0.078 020	6.885 203
23:30:00	0.925 703	0.105 673	9.685 247
23:35:00	1.452 955	0.210 169	20.240 73
23:40:00	39.424 84	0.956 483	95.605 67
23:45:00	27.292 74	0.963 866	96.353 23
23:50:00	18.360 96	0.969 226	96.895 32
23:55:00	66.133 92	0.999 568	99.956 46
0:00:00	48.123 30	0.986 940	98.683 40
0:05:00	21.367 40	0.912 350	91.148 93
0:10:00	14.654 56	0.941 284	94.072 71
0:15:00	9.929 699	0.958 296	95.791 50
0:20:00	9.634 705	0.942 674	94.218 59
0:25:00	6.664 675	0.959 603	95.925 83
0:30:00	2.897 037	0.876 591	87.537 38
0:35:00	1.427 190	0.717 862	71.504 99
0:40:00	0.770 491	0.497 582	49.256 48
0:45:00	1.196 706	0.663 697	66.036 84
0:50:00	1.472 002	0.792 386	79.035 31
0:55:00	0.885 855	0.594 668	59.062 67
1:00:00	1.292 329	0.612 853	60.904 97
1:05:00	1.558 680	0.755 496	75.311 38
1:10:00	1.929 264	0.760 412	75.813 52
1:15:00	2.274 060	0.763 764	76.157 74
1:20:00	2.165 221	0.858 157	85.684 74
1:25:00	2.489 559	0.851 919	85.060 09
1:30:00	2.787 737	0.845 126	84.379 63
1:35:00	1.369 624	0.688 572	68.549 51
1:40:00	1.753 171	0.699 683	69.677 37

通过 SDS-CF 的云取证流程验证实验可知，SDS-CF 能实现智能云取证服务.

4.4 实验 2.云取证趋势分析算法效果对比

为了验证基于隐 Markov 模型的云取证趋势分析算法的效果，本实验将对 3 种取证策略进行对比.一种是 3.5 节提出的云取证策略(cloud forensics strategy, CFS)，另外 2 种是实际取证分析中常用到的网络取证策略(network forensics strategy, NFS)和云计算平台事后取证策略(afterword forensics strategy, AFS).NFS 策略仅对网络流量进行实时取证，与 CFS 的主要区别在于在协同实时取证状态 s_3 和全量取证状态 s_4 下，仅对网络日志和网络流量进行取证.AFS 不对网络流量进行取证，仅在入侵发生后对虚拟机进行取证，与 CFS 主要区别在于在网络取证状态 s_2 下不予取证，在协同实时取证状况 s_3 和全量取证状态 s_4 下对虚拟机日志和虚拟机镜像进行取证.

1) CFS,NFS 和 AFS 参数确定

定义 5 提到的取证代价系数 $\boldsymbol{C}(i)$ 以取证时间窗口 T 内的取证存储开销为依据，从大量真实取证案例中统计得出.目前，网络取证业内广泛采用的取证时间窗口是 3 600 s.通过分析某司法鉴定所的案例数据可得出：在不取证状态，取证存储开销为 0 MB；在网络取证状态对网络设备日志进行提取，3 600 s 存储开销为 10 MB 级别；在协同实时取证状态下对网络设备日志、网络流量镜像、云计算平台日志、虚拟机日志进行提取，3 600 s 存储开销为 1 GB 级别；在全量取证状态下对网络设备日志、网络流量镜像、云计算平台日志、虚拟机镜像进行提取，存储开销为 100 GB 级别.不同的取证策略，其提取的数据对象不同，因而存储开销也不同.

CFS 的取证策略参数与取证代价系数取值为

$$\boldsymbol{D}=\begin{pmatrix}0&0&0&0&0\\1&0&0&0&0\\1&1&1&1&0\\1&1&1&1&1\end{pmatrix},\tag{21}$$

$$\boldsymbol{C}=(0\quad 0.01\quad 1\quad 100),\tag{22}$$

NFS 的取证策略参数与取证代价系数取值为

$$\boldsymbol{D}=\begin{pmatrix}0&0&0&0&0\\1&0&0&0&0\\1&1&0&0&0\\1&1&0&0&0\end{pmatrix},\tag{23}$$

$$\boldsymbol{C}=(0\quad 0.01\quad 1\quad 1),\tag{24}$$

AFS 的取证策略参数与取证代价系数取值为

$$\boldsymbol{D}=\begin{pmatrix}0&0&0&0&0\\0&0&0&0&0\\0&0&1&1&1\\0&0&1&1&1\end{pmatrix},\tag{25}$$

$$\boldsymbol{C}=(0\quad 0\quad 100\quad 100),\tag{26}$$

2) 实验对比分析

根据 3.2 节中提到的算法计算出观测时间段 T 内云取证状态概率 λ 分布如图 5 所示.

从图 5(a)中可以看出从攻击开始至 22:30:00 之间,云取证状态处于 s_1 的概率较大;从图 5(b)中可以看出 22:30:00—23:25:00 之间,云取证状态处于 s_2 的概率较大;从图 5(c)中可以看出 23:25:00 之后,大部分时间云取证状态处于处于 s_3 的概率较大,从图 5(d)中可以看出其中 23:45:00—00:00:00 之间云取证状态出现了 2 次概率为 s_4 状态的概率波峰.根据 4.3 节对 LLDOS1.0 的数据分析可知,图 5(b)的高峰与攻击过程的扫描探测阶段对应,图 5(c)的高峰与攻击过程的入侵阶段和后续的 DDoS 攻击对应,图 5(d)中的高峰与攻击过程的攻陷阶段对应.由此可见,云取证状态概率模型较准确地反映了攻击过程的云取证状态概率分布.

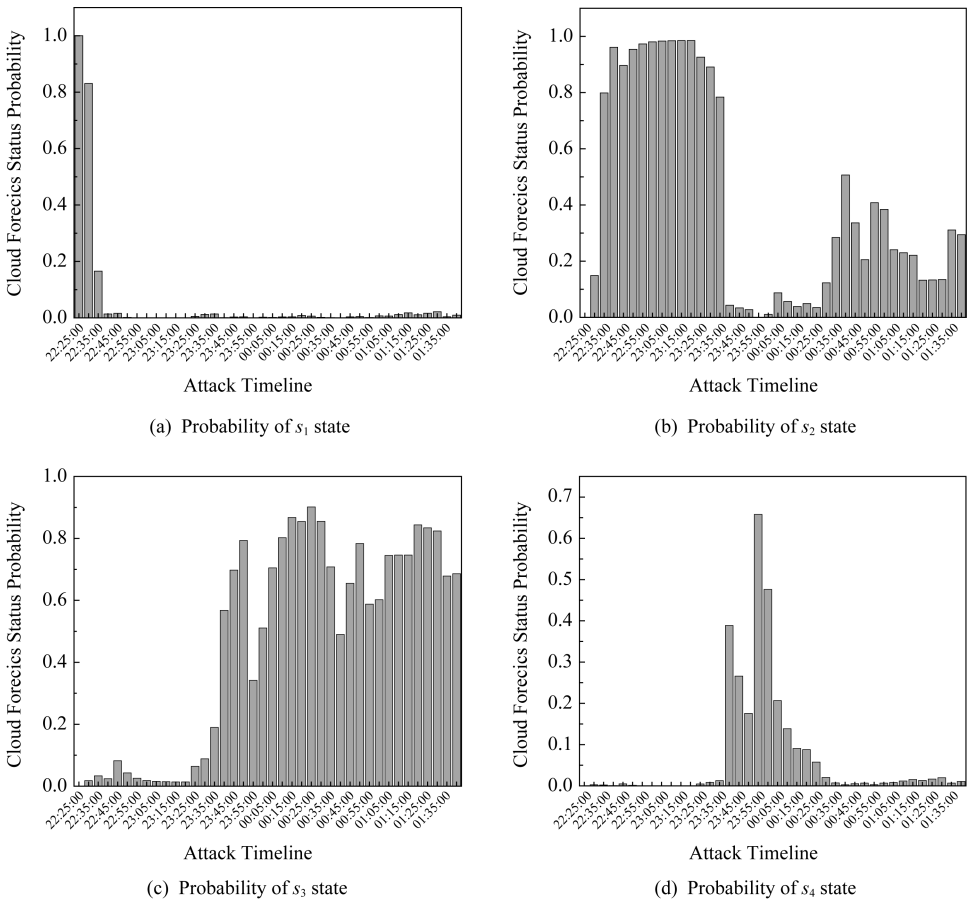


Fig. 5 Probability distribution of cloud forensics status during the attack process of LLDOS 1.0

图 5 LLDOS1.0 攻击过程云取证状态概率分布

根据 4.2 节、4.3 节中模型参数和 3.2 节提出的云取证趋势分析算法计算出 LLDOS1.0 攻击过程的 CFS,NFS 和 AFS 三种取证策略的云取证趋势分析结果如表 2 所示.将表 2 中的数据制成对比曲线图如图 6 所示,从图 6 可以看出,23:25:00 之前 CFS,NFS 和 AFS 三种策略的取证趋势均在低位波动,根据 4.3 节对 LLDOS1.0 的数据分析可知,此时网络攻击处于第 2 阶段,也就是扫描探测阶段,次阶段这 3 种策略对应的取证资源需求均小.23:30:00—00:00:00 之间,CFS 的取证趋势出现了 2 次波峰,此时攻击进入阶段 3,4,波峰对应着严重的入侵攻击事件和攻陷事件,CFS 的取证资源需求较大;NFS 的取证趋势持续保持在低位,因为该策略此阶段依然只进行网络日志和网络流量取证,取证资源需求较小;AFS 的取证趋势强势走高,是因为此阶段系统被入侵或者攻陷,进入事后应急响应过程,取证资源需求较大.

根据表 2 中的数据计算出 3 种取证策略的取证能力和取证开销 2 个量化指标如表 3 所示.

通过表 3 可以看出,CFS 策略取证能力为 91.6%,明显优于 NFS 的 68.9%和 AFS 的 50.5%.而取证开销方面 CFS 是 NFS 的 13 倍,是 AFS 的 13.8%,

介于两者之间.通过以上分析可知,NFS 策略没有考虑到网络入侵后云计算平台中虚拟机的取证需求,虽然取证开销小,但是取证能力不足,关键电子证据损失较多.AFS 策略因遗漏了实时网络取证需求,并对虚拟机进行粗粒度的取证策略导致取证能力较低而取证开销过高.而 SDS-CF 中采用的 CFS 策略能够在取证能力和取证开销方面有一个较好的平衡.

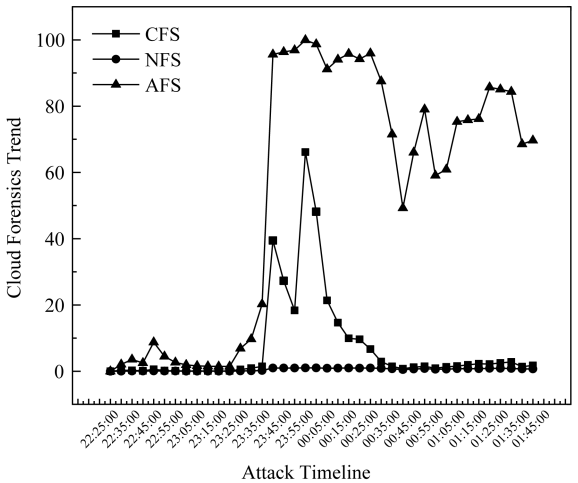


Fig. 6 Comparison of cloud forensics trend during the attack process of LLDOS1.0

图 6 LLDOS1.0 攻击过程云取证趋势对比

Table 3 Comparison of the Effectiveness of Three Forensics Policies Based on LLDOS1.0

表 3 基于 LLDOS1.0 的 3 种取证策略的效果对比

Forensics Strategy	*CoF*	*OoF*
CFS	0.915 844	292.733 3
NFS	0.689 272	21.377 6
AFS	0.505 128	2 121.348 0

5 总 结

本文提出一种基于软件定义安全和云取证趋势分析的智能云取证方法,该方法包括 1 个架构和 2 个核心算法.1)基于软件定义安全的云取证架构,通过该架构实现云网络与云计算平台实时协同取证,有效规避事后取证方法带来的种种难题,并解决云网络东西向流量的取证方法缺失和云用户网络取证手段缺失的问题;2)基于隐 Markov 模型的云取证趋势分析算法和基于改进告警质量的 IDS 告警选择算法,将该算法运用于基于软件定义安全的云取证架构中进行取证策略智能决策和取证资源智能调度,实现智能云取证,提高云取证准确度和效率.通过实验证明了该云取证方法的可行性,且该取证方法取证能力为 91.6%,明显高于网络取证和云计算平台事后取证方法,取证开销介于网络取证和事后取证之间,在取证能力和取证开销之间取得了较好的平衡.该方法对云服务商提供云取证服务具有广泛的借鉴意义.未来,一方面将继续优化云取证趋势分析算法,将攻击时长等影响因素考虑进来,进一步提高算法的准确性和降低取证开销;另一方面随着网络攻防技术和云计算的不断向前发展,网络攻防形式会不断发生变化,对应的模型参数需要不断地调整,下一步研究一种更加智能的模型参数训练方法.

参 考 文 献

[1] CNCERT. 2018 年我国互联网网络安全态势综述[EB/OL]. [2019-05-01]. http://www.cac.gov.cn/1124379080_15554834432651n.pdf

[2] Ding Liping, Wang Yongji. Study on relevant law and technology issues about computer forensics [J]. Journal of Software, 2005, 16(2): 260-275 (in Chinese)
(丁丽萍, 王永吉. 计算机取证的相关法律技术问题研究[J]. 软件学报, 2005, 16(2): 260-275)

[3] Ruan K, Carthy J, Kechadi T, et al. Cloud forensics [G] // IFIPAICT 361: Proc of the 7th IFIP WG 11.9 Int Conf on Digital Forensics. Berlin: Springer, 2011: 35-46

[4] Mai Yonghao, Zou Jinpei, Xu Rongsheng, et al. Computer Forensics and Judicial Expertise (3rd edition)[M]. Beijing: Tsinghua University Press, 2018: 139 (in Chinese)
(麦永浩, 邹锦沛, 许榕生, 等. 计算机取证与司法鉴定(第 3 版)[M]. 北京: 清华大学出版社, 2018: 139)

[5] Ding Liping, Liu Xuehua. The research of digital forensics in cloud computing [J]. China Information Security, 2019, 10(5): 59-60 (in Chinese)
(丁丽萍, 刘雪花. 云环境下的电子数据取证技术研究[J]. 中国信息安全, 2019, 10(5): 59-60)

[6] Khan S, Gani A, Wahab A W A, et al. Network forensics: Review, taxonomy, and open challenges [J]. Journal of Network and Computer Applications, 2016, 66: 214-235

[7] Khan S, Gani A, Wahab A W A, et al. Towards an applicability of current network forensics for cloud networks: A SWOT analysis [J]. IEEE Access, 2016, 4: 9800-9820

[8] Hunt R, Zeadally S. Network forensics: An analysis of techniques, tools, and trends [J]. Computer, 2012, 45(12): 36-43

[9] Jiang Liu, Tian Guiyan, Zhu Shidong. Design and implementation of network forensic system based on intrusion detection analysis [C] //Proc of the 1st Int Conf on Control Engineering and Communication Technology. Piscataway, NJ: IEEE, 2012: 689-692

[10] Chen Zhen, Han Fuye, Cao Junwei, et al. Cloud computing-based forensic analysis for collaborative network security management system [J]. Tsinghua Science and Technology, 2013, 18(1): 40-50

[11] Bon K. Integrating intrusion alert information to aid forensic explanation: An analytical intrusion detection framework for distributive IDS [J]. Information Fusion, 2009, 10(4): 325-341

[12] Dou Wanchun, Chen Qi, Chen Jinjun. A confidence-based filtering method for DDoS attack defense in cloud environment [J]. Future Generation Computer Systems—the International Journal of Grid Computing and eScience, 2013, 29(7): 1838-1850

[13] Li Jun, Sung Minho, Xu Jun, et al. Large-scale IP traceback in high-speed Internet: Practical techniques and theoretical foundation [C] //Proc of the 2004 IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2004: 115-129

[14] Mizoguchi S, Takemori K, Miyake Y, et al. Traceback framework against botmaster by sharing network communication pattern information [C] //Proc of the 5th Int Conf on Innovative Mobile and Internet Services in Ubiquitous Computing. Piscataway, NJ: IEEE, 2011: 639-644

[15] Kim H S, Kim H K. Network forensic evidence acquisition (NFEA) with packet marking [C] //Proc of the 2011 9th IEEE Int Symp on Parallel and Distributed Processing with Applications Workshops. Piscataway, NJ: IEEE, 2011: 388-393

[16] Yan Fen, Zhu Hui, Chen Shuangshuang, et al. A lightweight IP traceback scheme depending on TTL [J]. Procedia Engineering, 2012, 29: 1932-1937

[17] Gao Junfeng, Zhang Yuefeng, Lou Senlin, et al. Research on taint backtracking reverse analysis method of network encoding protocol [J]. Netinfo Security, 2017, 17(1): 68-76 (in Chinese)
(高君丰, 张岳峰, 罗森林, 等. 网络编码协议污点回溯逆向分析方法研究[J]. 网络信息安全, 2017, 17(1): 68-76)

[18] Jeong E, Lee B. An IP traceback protocol using a compressed hash table, a sinkhole router and data mining based on network forensics against network attacks [J]. Future Generation Computer Systems, 2014, 33(4): 42-52

[19] Monsanto C, Reich J, Foster N, et al. Composing software-defined networks [C] //Proc of the 10th USENIX Conf on Networked Systems Design and Implementation. Berkeley, CA: USENIX Association, 2013: 1-14

[20] Wei Zhanzhen, Wang Shourong, Li Zhaobin, et al. Research on SDN terminal access control based on openFlow [J]. Netinfo Security, 2018, 18(4): 29-37 (in Chinese)
(魏占祯, 王守融, 李兆斌, 等. 基于 OpenFlow 的 SDN 终端接入控制研究[J].网络信息安全, 2018, 18(4): 29-37)

[21] Francois J, Festor O. Anomaly traceback using software defined networking [C] //Proc of 2014 IEEE Int Workshop on Information Forensics and Security (WIFS). Piscataway, NJ: IEEE, 2014: 203-208

[22] Khan S, Gani A, Wahab A W A, et al. FML: A novel forensics management layer for software defined networks [C] //Proc of the 2016 6th Int Conf-Cloud System and Big Data Engineering (Confluence). Piscataway, NJ: IEEE, 2016: 619-623

[23] Achleitner S, Porta T L, Jaeger T, et al. Adversarial network forensics in software defined networking [C] //Proc of the Symp on SDN Research. New York: ACM, 2017: 8-20

[24] Amin H-F, Muthu R, Dilshad S. Strategic Engineering for Cloud Computing and Big Data Analytics [M]. Berlin: Springer, 2017: 189-205

[25] Mohiddin S K, Yalavarthi S B, Sharmila S. A complete ontological survey of cloud forensic in the area of cloud computing [C] //Proc of the 6th Int Conf on Soft Computing for Problem Solving. Berlin: Springer, 2017: 38-47

[26] Ruan K, Carthy J, Kechadi T, et al. Cloud forensics definitions and critical criteria for cloud forensic capability: An overview of survey results [J]. Digital Investigation the Int Journal of Digital Forensics, 2013, 10(1): 34-43

[27] Dykstra J, Sherman A T. Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques [J]. Digital Investigation, 2012, 9(8): 90-98

[28] Santra P, Roy A, Majumder K. A comparative analysis of cloud forensic techniques in IaaS [G] //AISC 554: Proc of ICCCCS 2016. Berlin: Springer, 2018: 207-215

[29] Xie Yalong, Ding Liping, Lin Yuqi, et al. ICFF: A cloud forensics framework under the IaaS model [J]. Journal on Communications, 2013, 34(5): 200-206 (in Chinese)
(谢亚龙, 丁丽萍, 林渝淇, 等. ICFF: 一种 IaaS 模式下的云取证框架[J]. 通信学报, 2013, 34(5): 200-206)

[30] Alluri B K S P K R, Geethakumari G. A digital forensic model for introspection of virtual machines in cloud computing [C] //Proc of the 2015 IEEE Int Conf on Signal Processing, 2015 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES). Piscataway, NJ: IEEE, 2015: 1-5

[31] Cui Chaoyuan, Li Yonggang, Wu Yun, et al. A memory forensic method based on hidden event trigger mechanism [J]. Journal of Computer Research and Development, 2018, 55(10): 2278-2290 (in Chinese)
(崔超远, 李勇钢, 乌云, 等. 一种基于隐藏事件触发机制的内存取证方法[J]. 计算机研究与发展, 2018, 55(10): 2278-2290)

[32] Ahsan M A M, Wahab A W A, Idris M Y I, et al. CLASS: Cloud log assuring soundness and secrecy scheme for cloud forensics [EB/OL]. [2019-06-01]. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8355685

[33] Zawoad S, Dutta A K, Hasan R. SecLaaS: Secure logging-as-a-service for cloud forensics [C] //Proc of the 8th ACM SIGSAC Symp on Information, Computer and Communications Security. New York: ACM, 2013: 219-230

[34] Liu Wenmao, Qiu Xiaofeng, Wang Xiang. Software defined security SDN/NFV network security disclosure [M]. Beijing: China Machine Press, 2016: 91-100 (in Chinese)
(刘文懋, 裘晓峰, 王翔. 软件定义安全 SDN/NFV 新型网络的安全揭秘[M]. 北京: 机械工业出版社, 2016: 91-100)

[35] Wang Jian, Zhao Guosheng, Zhao Zhongnan, et al. Formal modeling and factor analysis for vulnerability propagation oriented to SDN [J]. Journal of Computer Research and Development, 2018, 55(10): 2256-2268 (in Chinese)
(王健, 赵国生, 赵中楠, 等. 面向 SDN 的脆弱性扩散形式化建模与扩散因素分析[J]. 计算机研究与发展, 2018, 55(10): 2256-2268)

[36] Qiu Xiaofeng, Cheng Fangyuan, Wang Weijia, et al. A security controller-based software defined security architecture [C] //Proc of the 2017 20th Conf on Innovations in Clouds, Internet and Networks (ICIN). Piscataway, NJ: IEEE, 2017: 191-195

[37] Pan Chen, Liu Zhiqiang, Liu Zhen, et al. Research on scalability of blockchain technology: Problems and methods [J]. Journal of Computer Research and Development, 2018, 55(10): 2099-2110 (in Chinese)
(潘晨, 刘志强, 刘振, 等. 区块链可扩展性研究: 问题与方法[J]. 计算机研究与发展, 2018, 55(10): 2099-2110)

[38] Qiu Xiaofeng. A data driven orchestration framework in software defined security [C] //Proc of the 2016 IEEE Int Conf on Network Infrastructure and Digital Content (IC-NIDC). Piscataway, NJ: IEEE, 2017: 34-39

[39] Rabiner L R. A tutorial on hidden Markov models and selected applications in speech recognition [G] //Reading in Speech Recognition. San Francisco: Morgan Kaufmann, 1990: 267-297

[40] Ephraim Y, Merhav N. Hidden Markov processes [J]. IEEE Transactions on Information Theory, 2002, 48(6): 1518-1569

[41] Sendi A S, Dagenais M, Jabbarifar M. Real time intrusion prediction based on optimized alerts with hidden Markov model [J]. Journal of Networks, 2012, 7(2): 311-321

[42] Xi Rongrong, Yun Xiaochun, Zhang Yongzheng, et al. An improved quantitative evaluation method for network security [J]. Chinese Journal of Computers, 2015, 38(4): 749-758 (in Chinese)
(席荣荣, 云晓春, 张永铮, 等. 一种改进的网络安全态势量化评估方法[J]. 计算机学报, 2015, 38(4): 749-758)

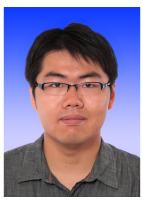
[43] Fatemeh K, Behzad A. Automatic learning of attack behavior patterns using Bayesian networks [C] //Proc of the 2012 6th Int Symp on Telecommunications (IST 2012). Piscataway, NJ: IEEE, 2012: 999-1004

[44] Lippmann R, Haines J W, Fried D J, et al. Analysis and results of the 1999 DARPA off-Line intrusion detection evaluation [J]. International Symp on Recent Advances in Intrusion Detection, 2000, 34(4): 162-182

Liu Xuehua, born in 1985. PhD candidate. Her main research interests include digital forensics, information security, cloud security.

Ding Liping, born in 1965. PhD, professor, PhD supervisor. Her main research interests include digital forensics, system security, covert channel analysis, privacy protection.

Liu Wenmao, born in 1983. PhD. His main research interests include cloud security, IoT security, threat intelligence and advanced security analytics.



Zheng Tao, born in 1986. Master. His main research interests include advanced 5G wireless communication technologies.

Li Yanfeng, born in 1984. PhD candidate. His main research interests include network covert channel, blockchain.

Wu Jingzheng, born in 1982. PhD, professor. His main research interests include system security, vulnerability mining, mobile security.